Proceedings

THIRTY-FIRST ANNUAL ALLERTON CONFERENCE ON COMMUNICATION, CONTROL, AND COMPUTING

September 29 - October 1, 1993

Allerton House, Monticello, Illinois
Sponsored by the
Coordinated Science Laboratory and the
Department of Electrical and Computer Engineering of the
University of Illinois at Urbana-Champaign

PROCEEDINGS

THIRTY-FIRST ANNUAL ALLERTON CONFERENCE ON COMMUNICATION, CONTROL, AND COMPUTING

Dilip V. Sarwate
Paul Van Dooren
Conference Co-Chairs

Conference held
September 29 - October 1, 1993
Allerton House
Monticello, Illinois

Sponsored by
The Coordinated Science Laboratory
and
The Department of Electrical and Computer Engineering
of the
UNIVERSITY OF ILLINOIS
at
Urbana-Champaign

NOTICE: **Return or renew all Library Materials! The *Minimum* Fee for each Lost Book is $50.00.**

The person charging this material is responsible for its return to the library from which it was withdrawn on or before the **Latest Date** stamped below.

Theft, mutilation, and underlining of books are reasons for disciplinary action and may result in dismissal from the University.
To renew call Telephone Center, 333-8400

UNIVERSITY OF ILLINOIS LIBRARY AT URBANA-CHAMPAIGN

ENGINEERING

L161—O-1096

UNIVERSITY OF ILLINOIS
LIBRARY
AT URBANA-CHAMPAIGN
ENGINEERING

621.3811
Al5
31(1993)

Engineering

Acknowledgments

This *Proceedings* contains papers presented at the Thirty-first Annual Allerton Conference on Communication, Control, and Computing which was held at Allerton House, Monticello, Illinois, on September 29, 30, and October 1, 1993. As in past years, the Conference was sponsored by the Coordinated Science Laboratory and the Department of Electrical and Computer Engineering of the University of Illinois at Urbana-Champaign. It is a pleasure to express our gratitude to Professor W. Kenneth Jenkins, Director of the Coordinated Science Laboratory, and to Professor Timothy N. Trick, Head of the Department of Electrical and Computer Engineering for their generous support and constant encouragement which contributed greatly to the success of the Conference.

Ninety-two invited papers and fifty-three contributed papers were presented at the Conference in twenty-six sessions, and we thank the authors for contributing to the success of the Conference. We are also very grateful to our colleagues at the University of Illinois at Urbana-Champaign: Professors B. Bamieh, T. Başar, T. Ü. Başar, B. Hajek, S. Hutchinson, D. L. Jones, P. R. Kumar, M. C. Loui, W. K. Jenkins, J. V. Medanic, S. P. Meyn, M. T. Orchard, W. R. Perkins, J. Ponce, M. B. Pursley (now with Clemson University), M. W. Spong, R. S. Sreenivas, M. E. Van Valkenburg, A. Vardy, P. G. Voulgaris, and K. A. Zeger for organizing many of the invited sessions, reviewing the contributed papers, chairing sessions, and helping in many other ways in the smooth running of the Conference. Invited sessions were also organized by Dr. D. E. Borth of Motorola, and Professors J. Y. Hui of Rutgers University, S. P. R. Kumar of Northwestern University, G. H. Sasaki of the University of Hawaii, S. Verdú of Princeton University, and B. F. Wyman of The Ohio State University, and we are especially grateful to them for their efforts on behalf of the Conference. Professors D. Brady of Northeastern University, R. G. Gallager of the Massachusetts Institute of Technology, S. A. Jordan of Northwestern University, and B. M. Waxman of Southern Illinois University chaired sessions at the Conference and for this we thank them.

We are grateful to Dr. Jose B. Cruz, Jr., Dean of Engineering at The Ohio State University, formerly a faculty member at the University of Illinois and a founder of this Conference, for attending the Conference and giving an after-dinner speech on "The Origins of the Allerton Conference." His reflections on how it all began and his encouraging words for the future of the Conference are greatly appreciated. We are also indebted to the Graduate Student Quartet of the University for providing musical entertainment at the Conference.

We thank Mrs Beverly Becker and her staff at Allerton House for cheerfully accommodating our Conference. Their constant attention to our needs contributed much to the success of the Conference. It is also a great pleasure to thank all the staff members of the Coordinated Science Laboratory who so ably handled all the practical aspects of organizing the Conference. In particular, we wish to mention Mr. Steven Williams and Mrs. Susan Baxley who handled all the business affairs, Mr. Robert Ebeling who printed large numbers of programs and registration packets, Mr. Robert MacFarlane who drafted maps to guide attendees, and Mr. Dan Jordan who handled the sales and shipping of these *Proceedings*. Most of all, however, we want to thank our Conference Secretary, Mrs. Janet Reese, who, with the able assistance of Ms Francie Bridges, Ms Dixie Murphy, and Ms Lila Rhoades, worked all year round in handling announcements, calls for papers, program preparation, Allerton House arrangements, registration, and the final editing of these Proceedings. The lion's share of the credit for the success of the Conference belongs to these ladies and it gives us the greatest pleasure to acknowledge this.

Dilip V. Sarwate

Paul Van Dooren

TABLE OF CONTENTS

ACKNOWLEDGMENTS i

I-A: MULTIPLE ACCESS COMMUNICATIONS

Chairman and Organizer: S. Verdú
Princeton University

ITERATIVE ALGORITHMS FOR JOINT DATA DETECTION AND DELAY ESTIMATION FOR CODE DIVISION MULTIPLE ACCESS COMMUNICATION SYSTEMS 1
A. Radović and B. Aazhang

ON CAPACITY AND SIGNATURE WAVEFORM DESIGN FOR SLOWING FADING CDMA CHANNELS 11
R. S. Cheng

SCHEDULED ACCESS WITH A SPACE COMPONENT 21
A. Ephremides

RANDOM ACCESS WITH LARGE PROPAGATION DELAY 24
R. Murali and B. Hughes

CHANNEL CODING FOR A FREQUENCY-HOPPED WIRELESS TRANSCEIVER 34
V. S. Lin and G. J. Pottie

LATTICE CODES FOR THE GAUSSIAN MULTIPLE-ACCESS CHANNEL 44
B. Rimoldi

SOFT-DECISION MULTISTAGE DETECTION FOR ASYNCHRONOUS AWGN CHANNELS 54
X. Zhang and D. Brady

I-B: INTELLIGENT CONTROL

Organizers: P. R. Kumar, T. Başar, and W. R. Perkins
University of Illinois at Urbana-Champaign
Chairman: P. R. Kumar
University of Illinois at Urbana-Champaign

HYBRID SYSTEM MODELING AND EVENT IDENTIFICATION 64
P. J. Antsaklis, M. Lemmon and J. A. Stiver

DIAGNOSING PROCESS FAULTS USING NEURAL NETWORK MODELS 74
K. L. Buescher, R. D. Jones, and M. J. Messina

ON BANDIT PROBLEMS WITH SIDE OBSERVATIONS AND LEARNABILITY 83
S. R. Kulkarni

FORMATION AND AGREEMENT PROBLEMS FOR ANONYMOUS MOBILE ROBOTS 93
I. Suzuki and M. Yamashita

I-C: ADVANCES IN COMMUNICATION, CONTROL, AND SIGNAL PROCESSING EDUCATION

Organizers: T. U Başar, and M. E. Van Valkenburg
University of Illinois at Urbana-Champaign
Chairman: T. U. Başar
University of Illinois at Urbana-Champaign

THE CHANGING MISSION OF ENGINEERING EDUCATION 103
C. F. Sechrist

MECHATRONIC SYSTEM DESIGN AT RENSSELAER 109
K. Craig

HIERARCHICAL CONTROL IN CIM EDUCATION 119
L. S.-B. King, J. Abell, and W. Erevelles

A SELF-PACED MODULAR COMPUTER-BASED DIGITAL SIGNAL PROCESSING LABORATORY COURSE 129
S. K. Mitra

WORKSTATIONS IN A COMMUNICATION SYSTEMS LABORATORY 130
B. A. Black

TEACHING PRINCIPLES AND EXPERIMENTATIONS OF TELECOMMUNICATIONS NETWORKS 140
J. Y. Hui

I-D: COMMUNICATION NETWORKS

Chairman: M. Waxman
Southern Illinois University

UPPER BOUND FOR THE CAPACITY OF A HYBRID RANDOM ACCESS SYSTEM WITH MULTI-USER CODING 149
R. M. Al-Rumaih and P. Mathys

SCHEDULING BROADCAST COMMUNICATION ON A HIGH-SPEED NETWORK 157
M. Valerio, L. E. Moser, and P. M. Melliar-Smith

DELAY ANALYSIS OF A CIRCUIT-SWITCHED INTERCONNECTION NETWORK IMPLEMENTING A GATED HOLD STRATEGY 167
P. Dietrich and R. R. Rao

THROUGHPUT PROPERTIES OF FAIR POLICIES IN RING NETWORKS 177
L. Georgiadis, R. Guerin and I. Cidon

ACCESS CONTROL TO TWO MULTISERVER LOSS QUEUES IN SERIES 187
C.-Y. Ku and S. Jordan

STRICT PEAK RATE ENFORCEMENT AT THE ATM NETWORK ENTRY POINT 197
J. Brassil

WHY CHANNEL ASSIGNMENT ALGORITHMS ARE ASYMPTOTICALLY SMART 207
R. J. McEliece, E. R. Rodemich and S. K. Deora

II-A: NUMERICAL METHODS

Organizer and Chairman: P. Van Dooren
University of Illinois at Urbana-Champaign

TRANSFER FUNCTION AND STATE SPACE APPROACHES TO RATIONAL MATRIX FACTORIZATION 217
M. S. Verma

MODEL REDUCTION VIA STATE SPACE PERTURBATIONS 227
H. W. Beke and D. Boley

LINEAR MATRIX INEQUALITIES IN SYSTEM AND CONTROL THEORY 237
S. Boyd, L. El Ghaoui, E. Feron and V. Balakrishnan

COMPUTING THE POLES OF AUTOREGRESSIVE MODELS FROM THE REFLECTION COEFFICIENTS 247
G. S. Ammar, D. Calvetti and L. Reichel

COMPUTATION OF THE STABLE DEFLATING SUBSPACE OF A SYMPLECTIC PENCIL USING STRUCTURE PRESERVING ORTHOGONAL TRANSFORMATIONS 255
R. V. Patel

EXPERIMENTAL EVALUATION OF THE GRADIENT ANNEALING ALGORITHM 265
M. Mohandes and C. Codrington

LEAST SQUARES APPROXIMATE SOLUTION OF OVERDETERMINED SYLVESTER EQUATIONS 275
A. S. Hodel

II-B: MOBILE AND PERSONAL COMMUNICATION SYSTEMS AND NETWORKS

Organizers: D. E. Borth and M. B. Pursley
Motorola, Inc. and Clemson University
Chairman: M. B. Pursley
Clemson University

UPLINK PERFORMANCE OF A CDMA WIRELESS COMMUNICATIONS SYSTEM 277
L. F. Chang

OPTIMAL RECEIVER PERFORMANCE FOR COMMUNICATION OVER A RAYLEIGH FADING CHANNEL WITH FADED INTERFERENCE 287
F. Ling

ADVANCED CDMA BASED ON CO-CHANNEL INTERFERENCE CANCELLATION 297
R. Kohno

WIRELESS DATA COMMUNICATIONS 307
K. Pahlavan

DOWN-LINK CAPACITY CONSIDERATIONS WHEN MULTIPLE SATELLITE BASED CDMA SYSTEMS WITH ORTHOGONAL SPREADING SHARE THE SAME SPECTRUM 316
B. R. Vojcic, R. L. Pickholtz, and L. B. Milstein

SLOW FREQUENCY-HOPPED CDMA FOR HIGH-MOBILITY PERSONAL COMMUNICATION SYSTEMS 325
P. D. Rasky, G. M. Chiasson, and D. E. Borth

II-C: HIGH SPEED NETWORKS AND SWITCHING

Chairman and Organizer: J. Hui
Rutgers University

NONBLOCKING MULTI-CHANNEL SWITCHING 335
H. Saidi, P. S. Min and M. V. Hegde

COD: ARCHITECTURES FOR OPTICAL TIME-BASED MULTIPLEXERS 345
R. L. Cruz and J. T. Tsai

MULTIPLEXING GAINS IN BITSTREAM MULTIPLEXORS 355
S. Jordan and I. Sidhu

AN ATM SWITCH WITH MULTI-STAGE CONTENTION RESOLUTION 365
J. Hui and T.-H. Lee

QUEUE MANAGEMENT WITH MULTIPLE DELAY AND LOSS PRIORITIES FOR ATM SWITCHES 375
H. J. Chao and I. H. Pekcan

ROUTING ALGORITHMS IN MULTICAST SWITCHING TOPOLOGIES 385
E. W. Zegura

A REARRANGEABLE ALGORITHM FOR MULTIPOINT ROUTING 395
B. M. Waxman

II-D. QUEUEING NETWORKS

Organizers: P. R. Kumar and S. P. Meyn
University of Illinois at Urbana-Champaign
Chairman: S. P. Meyn
University of Illinois at Urbana-Champaign

COMPUTABLE CONVERGENCE RATES AND GEOMETRIC ERGODICITY FOR MARKOVIAN QUEUEING SYSTEMS 404
R. L. Tweedie

MACROSCOPIC APPROXIMATIONS FOR A CLASS OF POINT PROCESSES AND STATE-DEPENDENT NON-MARKOVIAN QUEUEING NETWORKS 413
T. Konstantopoulos

FINITE MOMENTS FOR INVENTORY PROCESSES 423
K. Sigman and D. D. Yao

OPTIMAL CONTROL OF MANUFACTURING SYSTEMS WITH PRODUCTION DEPENDENT MACHINE FAILURES 433
G.-X. Yu and P. Vakili

FINITE HORIZON PRODUCTION PLANNING OF STOCHASTIC MANUFACTURING SYSTEMS: OPTIMALITY, TRACEABLITY AND TURNPIKE SETS 443
Q. Zhang and G. Yin

II-E: VIDEO CODING AND IMAGE PROCESSING

Chairman: M. T. Orchard
University of Illinois at Urbana-Champaign

HUMAN VISUAL SYSTEM BASED SUBBAND IMAGE COMPRESSION 452
T. P. O'Rourke and R. L. Stevenson

EVALUATION OF A FUNCTION OCCURRING IN MAXIMUM-LIKELIHOOD IMAGE-RESTORATION FOR CCD CAMERA DATA 462
D. L. Snyder, C. W. Helstrom, A. D. Lanterman, and M. Faisal

EFFICIENT SCAN PATTERNS FOR IMAGE DECORRELATION 463
N. D. Memon, K. Sayood, and S. Magliveras

RECOVERY OF LOW BIT RATE COMPRESSED IMAGES USING GIBBS PRIORS 473
T. Özçelik, J. C. Brailean, and A. K. Katsaggelos

BLOCK PROCESSING---AN UNSUPERVISED NON-MODEL-BASED IMAGE ANALYSIS APPROACH 483
T. Lei, Z. Zhao, and W. Sewchand

PARAMETER ESTIMATION FOR THE CURVE RECOVERY PROBLEM 485
B. E. Schmitz and R. L. Stevenson

III-A: ADVANCES IN ROBUST CONTROL

Organizers: B. Bamieh and T. Başar
University of Illinois at Urbana-Champaign
Chairman: B. Bamieh
University of Illinois at Urbana-Champaign

ALTERNATING PROJECTION METHODS FOR MIXED H^2 AND H^∞ NEHARI PROBLEMS 495
A. E. Frazho, K. M. Grigoriadis, and S. M. Kherat

H_2 AND H_∞ OPTIMAL REALIZATIONS OF FINITE WORDLENGTH DIGITAL FILTERS 505
M. A. Rotea and D. Williamson

ELIMINATING POLE/ZERO CANCELLATIONS IN MIMO H^∞/H^2 CONTROL USING INTERPOLATION 515
K. C. Goh and M. G. Safonov

LOW-ORDER CONTROL DESIGN FOR LMI PROBLEMS USING ALTERNATING PROJECTION METHODS 525
K. M. Grigoriadis and R. E. Skelton

NUMERICAL ISSUES IN COMPUTATION OF H^∞ CONTROLLERS FOR DISTRIBUTED PLANTS 535
O. Toker and H. Özbay

III-B: OPTICAL NETWORKS

Organizer and Chairman: G. H. Sasaki
University of Hawaii

WAVELENGTH ROUTING FOR ALL-OPTICAL NETWORKS 545
R. A. Barry and P. A. Humblet

COMPARISON OF FEED-FORWARD AND FEEDBACK CONFIGURATIONS OF THE *STAGGERING SWITCH* ARCHITECTURE 556
Z. Haas

SPACE-TIME NETWORKS FOR OPTICAL COMMUNICATIONS 566
A. V. Ramanan, H. F. Jordan, and J. R. Sauer

QUEUEING DYNAMICS AND THROUGHPUT OF RING NETWORKS WITH SPATIAL REUSE 576
N. Bambos and A. Nguyen

OPERATION OF NON-BUS-ORIENTED SINGLE-HOP INTERCONNECTIONS: BUSY-TONE MULTIPLE ACCESS AND FOLDED SCHEDULES 588
Y. Birk and N. Bloch

SCHEDULING TRANSMISSIONS IN WDM BROADCAST-AND-SELECT NETWORKS................... 598
G. R. Pieris and G. H. Sasaki

III-C: CODING THEORY

Chairman: A. Vardy
University of Illinois at Urbana-Champaign

A CLASS OF NON-LINEAR BLOCK CODES OVER Z_{2_m} FOR THE LEE METRIC 608
A. C. F. Pessoa and D. S. Arantes

PROPERTIES OF THE FOURIER TRANSFORM DISTILLED FROM ALGEBRAIC-GEOMETRY CODES 617
R. E. Blahut

d-FORM SEQUENCES: FAMILIES OF SEQUENCES WITH LOW CORRELATION VALUES AND LARGE LINEAR SPAN 625
A. Klapper

PARTIAL PERIOD CROSS-CORRELATIONS OF GEOMETRIC SEQUENCES 635
A. Klapper

PARTIAL EXPANSION PROPERTIES OF LINEAR BLOCK CODES 643
A. S. Khayrallah

BIT-LEVEL PARALLEL DECODING OF REED-SOLOMON CODES 653
S. Ray-Chaudhuri and A. H. Chan

SEQUENTIAL DECODING ON FINITE STATE CHANNELS WITHOUT INTERSYMBOL INTERFERENCE UNDER THE PE-CRITERION 655
I. Markman and J. B. Anderson

III-D: DISCRETE EVENT DYNAMIC SYSTEMS I

Organizer and Chairman: R. S. Sreenivas
University of Illinois at Urbana-Champaign

DES SUPERVISOR DESIGN FOR HYBRID CONTROL SYSTEMS 657
J. A. Stiver and P. J. Antsaklis

A NEW APPROACH FOR SOME COMBINATORIALLY HARD STOCHASTIC OPTIMIZATION PROBLEMS 667
C. G. Cassandras and V. Julka

PID-LIKE CONTROL OF DISCRETE EVENT SYSTEMS HAVING A DISCRETE STATE MODEL REPRESENTATION 677
R. A. DeCarlo, S. Drakunov, K. Passino, and U. Özgüner

OPTIMAL SENSOR AND ACTUATOR CHOICES FOR DISCRETE EVENT SYSTEMS 687
S. D. Young and V. K. Garg

CONTROL SYNTHESIS USING PATH ALGEBRAS FOR FORBIDDEN STATE AVOIDANCE 697
L. E. Holloway

NON-BLOCKING CONTROL OF NONDETERMINISTIC SYSTEMS 707
R. Kumar and M. A. Shayman

III-E: QUANTIZATION

Chairman: K. Zeger
University of Illinois at Urbana-Champaign

SEQUENTIAL SCALAR QUANTIZATION OF VECTORS 717
R. Balasubramanian, C. A. Bouman, and J. P. Allebach

OPTIMAL BUFFER CONTROL FOR VARIABLE-RATE LOSSY COMPRESSION 727
D. Tse, R. Gallager, and J. Tsitsiklis

NONPARAMETRIC DETECTION APPLIED TO VQ-BASED SPEAKER VERIFICATION 737
M. A. Lund and C. C. Lee

III-F: DETECTION THEORY

Chairman: D. Brady
Northeastern University

EFFECT OF CHANGE IN NOISE DISTRIBUTION ON DISTRIBUTED HYPOTHESIS TESTING SYSTEMS 739
S. I. Akkihal and G. L. Wise

QUICKEST DETECTION FOR DISTRIBUTED SENSOR SYSTEMS 748
R. W. Crow and S. C. Schwartz

A LOWER BOUND ON THE PROBABILITY OF ERROR IN MULTIHYPOTHESIS TESTING 758
H. V. Poor and S. Verdú

IV-A: SURVEYS IN HIGH-PERFORMANCE COMBINATORIAL ALGORITHMS

Organizer: B. Hajek
University of Illinois at Urbana-Champaign
Chairman: R. G. Gallager
Massachusetts Institute of Technology

PROGRESS TOWARD OPTIMAL NONBLOCKING MULTIPOINT VIRTUAL CIRCUIT SWITCHING NETWORKS 760
J. S. Turner

THE DESIGN AND ANALYSIS OF ELECTRONIC AND OPTICAL HIGH-PERFORMANCE PACKET (ATM) SWITCHES 770
M. J. Karol

SYSTEMS ISSUES IN MULTI-WAVELENGTH OPTICAL NETWORKS 780
R. Ramaswami

IV-B: ADVANCES IN NONLINEAR CONTROL

Organizers: B. Bamieh and T. Başar
University of Illinois at Urbana-Champaign
Chairman: T. Başar
University of Illinois at Urbana-Champaign

DESIGN OF ROBUST NONLINEAR FEEDBACK CONTROLS: CONCEPTS, TOOLS, AND PROCEDURES 791
R. A. Freeman and P. V. Kokotovic

DISTURBANCE ATTENUATION WITH STEADY-STATE CONSTRAINTS FOR NONLINEAR SYSTEMS 801
A. Isidori and T. J. Tarn

DISCRETE TIME SLIDING MODE CONTROL IN NONLINEAR SAMPLED-DATA SYSTEMS 809
W.-C. Su, Ü. Özgüner, S. V. Drakunov

DYNAMICAL SYSTEMS APPROACH TO VISION BASED RANGING FROM FEATURE OBSERVATIONS 819
M. Jankovic and B. K. Ghosh

IV-C: CDMA SYSTEMS

Chairman: D. V. Sarwate
University of Illinois at Urbana-Champaign

SIGNAL DESIGN FOR L-OUT-OF-K CDMA CHANNELS 829
A. A. Alsugair and R. S. Cheng

TRELLIS-BASED MULTIUSER RECEIVERS FOR CONVOLUTIONALLY CODED CDMA SYSTEMS 839
T. R. Giallorenzi and S. G. Wilson

GROUP DETECTION FOR SYNCHRONOUS CDMA COMMUNICATION OVER FREQUENCY-SELECTIVE FADING CHANNELS 849
M. K. Varanasi

IV-D: PERSONAL COMMUNICATION SYSTEMS AND NETWORKS

Organizer: S. P. R. Kumar
Northwestern University
Chairman: S. Jordan
Northwestern University

THE HIGHWAY PALM: A STOCHASTIC MODEL TO CAPTURE SPACE AND TIME DYNAMICS IN WIRELESS COMMUNICATION SYSTEMS 859
W. A. Massey and W. Whitt

ON THE CAPACITY BOUNDS IN CELLULAR SYSTEMS 868
Z. J. Haas, J. H. Winters and D. S. Johnson

DESIGN OPTIONS AND PERFORMANCE IN CELLULAR SYSTEMS 878
Y. Argyropoulos, S. P. R. Kumar and H. Tawfik

PERFORMANCE LIMITS FOR CELLULAR SYSTEMS WITH HANDOFFS 887
K. N. Sivarajan

IV-E: DISCRETE EVENT DYNAMIC SYSTEMS II

Organizer and Chairman: R. S. Sreenivas
University of Illinois at Urbana-Champaign

CONTROL OF PARTIALLY OBSERVED DISCRETE EVENT SYSTEMS WITH VARIABLE LOOKAHEAD MAXIMAL POLICIES 898
N. B. Hadj-Alouane, S. Lafortune, and F. Lin

DYNAMICS OF A SWITCHED SERVER SYSTEM UNDER CORRIDOR POLICIES 908
G.-X. Yu and P. Vakili

BOUNDS FOR THE BIAS AND VARIANCE IN REGENERATIVE SIMULATIONS VIA LORDEN'S INEQUALITY 918
P.-H. Chen and M. A. Zazanis

ON THE COMPUTATION OF NORMAL LANGUAGES 928
M. Barbeau, G. Custeau, and R. St-Denis

CONTROLLED STOCHASTIC LANGUAGES 938
H. Mortazavian

ON ACCELERATION OF STOCHASTIC APPROXIMATION FOR OPTIMIZATIONS OF DISCRETE EVENT SYSTEMS 948
I.-J. Wang and E. K. P. Chong

LIKELIHOOD RATIO METHODS VIA CONDITIONAL MONTE CARLO AND SPLITTING 958
M. C. Fu, J.-Q. Hu, and L. Shi

IV-F: STATISTICAL SIGNAL PROCESSING FOR EQUALIZATION AND TRACKING IN COMMUNICATION SYSTEMS

Chairman: D. L. Jones
University of Illinois at Urbana-Champaign

ERROR RATE AND MMSE PERFORMANCE OF LINEAR EQUALIZERS IN THE PRESENCE OF ADJACENT CHANNEL INTERFERENCE 968
M. Tahernezhadi and J. R. Cruz

BAYESIAN TECHNIQUES FOR KNOWN CHANNEL DECONVOLUTION 977
G.-K. Lee, S. B. Gelfand, and M. P. Fitz

ANALYSIS OF FREQUENCY-DEPENDENT PERFORMANCE OF LINEAR PREDICTOR FREQUENCY ESTIMATORS 987
R. M. Todd and J. R. Cruz

MAXIMUM LIKELIHOOD BASED ALGORITHM FOR MULTIPLE TARGET TRACKING 997
A. Satish and R. L. Kashyap

ON ADAPTIVE PHASE ALIGNMENT OF AN APERTURE ARRAY FOR OPTICAL SATELLITE-LINK COMMUNICATIONS 1007
M. Mehta, J. M. Morris, and M. Buckman

V-A: ROBOTICS AND COMPUTER VISION

Organizers: J. Ponce and S. Hutchinson
University of Illinois at Urbana-Champaign
Chairman: S. Hutchinson
University of Illinois at Urbana-Champaign

RECOVERING EXTRUDED SURFACES FROM A SINGLE IMAGE 1009
D. A. Forsyth and C. A. Rothwell

FORCE-GUIDED ASSEMBLY 1019
M. Peshkin, A. Goswami, and J. Schimmels

DERIVATION OF DEPTH MAPS THROUGH CONTROLLED ACTIVE EXPLORATION 1029
C. E. Smith and N. P. Papanikolopoulos

RECONSTRUCTING SURFACES FROM PROFILES: EPIPOLAR CURVES 1039
R. Szeliski and R. Weiss

V-B: CONTROL SYSTEMS

Chairman: P. Voulgaris
University of Illinois at Urbana-Champaign

STABILIZATION OF TIME-DELAY CONTROL SYSTEMS USING CAUCHY MATRIX 1048
A. S. C. Sinha

STABILITY ANALYSIS OF SYSTEMS WITH STATE SATURATION AND PARAMETER UNCERTAINTIES 1056
D. Liu, A. N. Michel, and K. Wang

ON STABILITY OF TIME-DELAY SYSTEMS 1058
G. M. Schoen and H. P. Geering

NECESSARY AND SUFFICIENT CONDITIONS FOR THE HURWITZ STABILITY OF INTERVAL MATRICES OF ARBITRARY ORDER WITH KNOWN ALGORITHMIC COMPLEXITY FOR THIRD ORDER 1061
H. Ye, K. Wang, and A. N. Michel

EXACT SLOW-FAST DECOMPOSITION OF THE H_∞ CONTROL RICCATI EQUATION OF SINGULARLY PERTURBED SYSTEMS 1071
A. H. Azzo and M. E. Sawan

ON A FINITE HORIZON H_∞ PERFORMANCE ROBUSTNESS PROBLEM 1073
M. B. Subrahmanyam

L_2 TO L_∞ GAINS FOR SAMPLED-DATA SYSTEMS 1077
G. Zhu and R. E. Skelton

SIMULTANEOUS AND INTERVAL-PLANT STABILIZATION IN DISCRETE-TIME 1079
E. Yaz

V-C: CONTROL OF COMMUNICATION NETWORKS

Chairman: B. Hajek
University of Illinois at Urbana-Champaign

OPTIMAL MULTIPLEXING ON A SINGLE LINK: DELAY AND BUFFER REQUIREMENTS 1081
L. Georgiadis, R. Guerin, and A. Parekh

ROBUST ROUTING FOR MAXIMUM WORST-CASE EXPECTED THROUGHPUT IN UNRELIABLE NETWORKS 1091
R. Ogier and V. Rutenburg

ADAPTIVE BACK-PRESURE CONGESTION CONTROL BASED ON LOCAL INFORMATION 1099
L. Tassiulas

STABILITY ANALYSIS OF SCHEDULING POLICIES IN RING NETWORKS WITH SPATIAL REUSE 1109
L. Georgiadis, W. Szpankowski and L. Tassiulas

STRUCTURAL PROPERTIES OF THE THROUGHPUT IN CYCLIC NETWORKS WITH GENERAL BLOCKING AND STARVATION 1119
R. Rajan and R. Agrawal

ADAPTIVE LOAD BALANCING IN NETWORKS 1129
E. Gelenbe and X. Mang

V-D: RECENT ADVANCES IN ADAPTIVE FILTERING

Organizer and Chairman: W. K. Jenkins
University of Illinois at Urbana-Champaign

LMS CONVERGENCE AND PERFORMANCE WITH A FILTERED-ERROR UPDATE 1139
J. B. Kenney, C. E. Rohrs, and V. P. Telang

EFFECTS OF QUASI-PERIODIC TRAINING SIGNALS ON ACOUSTIC ECHO-CANCELLER PERFORMANCE 1149
S. McCaslin and N. van Bavel

MODEL SET SELECTION FOR COMPLEXITY REDUCTION IN ADAPTIVE FILTERS 1156
S. Zimmermann and G. A. Williamson

A HIGH-PERFORMANCE COST-EFFECTIVE POLE-ZERO MMSE-DFE 1166
N. Al-Dhahir, A. Sayed, and J. Cioffi

FAST ALGORITHMS FOR DESIGNING STACK FILTERS 1176
J. Yoo, K. L. Fong, E. J. Coyle, and G. B. Adams III

V-E: SOME ANALYTIC METHODS WITH AN ALGEBRAIC TWIST

Organizer and Chairman: B. F. Wyman
Ohio State University

AN ALGEBRAIC APPROACH TO NULL-POLE COUPLING OPERATORS 1186
M. Rakowski and B. F. Wyman

A FRESH LOOK AT INVERSE DYNAMICAL SYSTEMS 1196
M. K. Sain

THE GLOBAL CONTROLLABILITY FILTRATION FOR IMPROPER TRANSFER FUNCTIONS AND THE WEDDERBURN-FORNEY CONSTRUCTION 1206
S. J. Giust

FACTORIZATION OF $(\sum_2,\sum_1)$-LOSSLESS FUNCTIONS: THE CONTINUOUS TIME CASE 1216
M. Rakowski

INDEX OF AUTHORS 1226

Iterative Algorithms for Joint Data Detection and Delay Estimation for Code Division Multiple Access Communication Systems

Andrej Radović
Communication Technology Laboratory
Swiss Federal Institute of Technology
ETH-Zentrum, Sternwartstrasse 7
CH-8092 Zürich, Switzerland

Behnaam Aazhang
Department of Electrical and
Computer Engineering
Rice University
Houston, Texas 77251-1892

Abstract

A multiuser communication system is considered where K users share an additive white Gaussian channel using the code division multiple access (CDMA) scheme. The main task of the study is the development of algorithms for joint data detection and delay estimation for multiuser systems. Since the maximum likelihood (ML) solution is very difficult to obtain, we rely on iterative algorithms to decompose the problem into K simpler optimization problems. A scheme is implemented where the Expectation Maximization (EM) algorithm estimates carrier phases, amplitudes and delays while a multistage detection algorithm detects the data. The performance of the resulting multiuser receiver is evaluated via Monte Carlo simulations. Results on the average bit-error probability prove that the proposed system is capable of eliminating the near-far effect which exists in conventional CDMA systems.

1 Introduction

In the last few years spread spectrum communications found application in commercial radio networking. Code division multiple access (CDMA) communications has now been accepted as a viable alternative to both time and frequency division schemes. In particular, CDMA has been shown to be effective for cellular mobile radio networks. The main difficulty in implementing CDMA for cellular systems is the near-far problem where strong signals interfere with the demodulation of weak signals either at the base station or the mobile units. This signal separation problem, when viewed from the perspective of the receiver, has been referred to as multiuser signal detection.

The optimum multiuser demodulation scheme was first developed by Verdu [1] and the work was followed by a number of simpler and sub-optimum strategies [2, 3, 4, 5, 6]. All of the proposed multiuser detection schemes exploit the knowledge of the signals of the interfering users to achieve, often substantial, performance improvement over the conventional matched filter detectors. Therefore, the performance of multiuser detectors is, in general, sensitive to the accuracy of the estimates of the interfering signals' parameters. In this paper, we address the sensitivity issues of the multiuser detection and investigate joint data demodulation and parameter estimation in the context of multiuser systems. In particular, we apply the expectation maximization (EM) and

---

This work was supported in part by the the Advanced Technology Program of the Texas Higher Education Coordinating Board under Grant 003604-018.

multistage techniques to develop iterative maximum likelihood algorithms for carrier phase, amplitude and delay estimation as well as data detection of multiple signals.

2 System Description

The code division multiple access system considered in this paper consists of multiple transmitters and receivers sharing an additive white Gaussian channel. It is assumed that K data packets are simultaneously transmitted over the channel using binary phase shift keying modulation and direct sequence spread spectrum signaling. Therefore, the base-band complex envelope of the transmitted signal of the k^{th} user is given as

$$s_k(t) = \sum_{i=-P}^{P} \sqrt{E_k/T}\, b_k^{(i)} a_k(t - iT) e^{j\theta_k}, \tag{1}$$

where $2P+1$ is the length of the data packet, $b_k^{(i)} \in \{-1,+1\}$ is the i^{th} data bit of the k^{th} user, and where $a_k(t) = \sum_{n=0}^{N-1} a_k^{(n)} \Pi_{T_c}(t - nT_c)$ is the time-limited spreading waveform assigned to the k^{th} user with spreading code symbol $a_k^{(n)} \in \{-1,+1\}$. In this model, $\Pi_{T_c}(t)$ denotes a unit rectangular pulse over the chip period $[0, T_c)$, E_k is the energy of the signal of the k^{th} user at the given receiver, T is the data period, $T_c = T/N$ is the chip period, and θ_k is the phase angle. We consider the receiver that is interested in the data packet of user 1. The signal at the receiver in this asynchronous multiuser system is the superposition of the K transmitted signals and noise, that is,

$$r(t) = \sum_{k=1}^{K} s_k(t - \tau_k) + n_t, \tag{2}$$

where τ_k is the relative delay of the k^{th} signal with respect to the receiver of interest and n_t is an observation of the complex white Gaussian noise process N_t with zero mean and power spectral density with height $\mathcal{N}_0/2$ for both the real and imaginary parts.

The problem of interest is demodulation of the data packet of user 1 (that is $b_1^{(i)}$ for $i = -P, \ldots, P$) when the delay τ_1, the amplitude $\sqrt{E_k/T}$ as well as the received phase angle $\phi_1 = \theta_1 - 2\pi f_c \tau_1$ of the modulated signal are unknown (f_c denotes the carrier frequency). In this model, it is appropriate to define a complex variable $\alpha_k = \sqrt{E_k/T} e^{j\phi_k}$ to represent both phase angle and the possibly unknown received signal amplitude. This problem is more complicated than the traditional BPSK demodulation considering that the parameters of the desired signal are unknown and that there are $K-1$ interfering signals. In the past, data demodulation and synchronization have always been decoupled where delay- and phase-locked loops provide timing and phase angle information to the matched filter for data detection. In this paper, we investigate the joint data detection and channel estimation problem where we can also exploit the knowledge of the spreading waveforms of the interfering users. Therefore, the task at hand is a joint multiuser detection and estimation problem with the log-likelihood function given as[1]

$$\Lambda(\mathcal{B}, \underline{\tau}, \underline{\alpha}) = \frac{2}{\mathcal{N}_0} \int \Re\{r(t) \mathcal{S}^*(t; \mathcal{B}, \underline{\tau}, \underline{\alpha})\}\, dt - \frac{1}{\mathcal{N}_0} \int |\mathcal{S}(t; \mathcal{B}, \underline{\tau}, \underline{\alpha})|^2\, dt, \tag{3}$$

---

[1] The notations $\Re\{X\}$ and X^* denote the real part and the conjugate of the complex variable X, and $\underline{X}^T$ denotes the transpose of the vector $\underline{X}$.

where $\mathcal{S}(t;\mathcal{B},\underline{\tau},\underline{\alpha}) = \sum_{k=1}^{K} s_k(t-\tau_k)$, the data bits are presented in a matrix form $\mathcal{B} = [b_k^{(i)}]_{K\times(2P+1)}$, and where $\underline{\tau} = (\tau_1,\ldots,\tau_K)^T$ and $\underline{\alpha} = (\alpha_1,\ldots,\alpha_K)^T$. The maximum likelihood solution of the joint detection and estimation problem would be the point in the parameter space where Λ achieves its maximum with respect to $\mathcal{B}$, $\underline{\tau}$, and $\underline{\alpha}$. This optimum scheme is computationally intensive and impractical considering typical data rates for CDMA systems. It is easy to see from the log-likelihood equation that the sufficient statistics are the outputs of K filters matched to the signals $s_k(t)$, $k = 1,\ldots,K$ observed for the entire length of data packets even if we consider estimating only τ_1 and α_1 and demodulating $b_1^{(i)}$.

3 Iterative Detection and Estimation Algorithm

The maximization of the generalized log-likelihood function (3) with respect to the data bits of the K users is impractical since we need to maximize the expected value of the log-likelihood function over $\underline{\tau}$ and $\underline{\alpha}$. Therefore, it is more feasible to resort to iterative algorithms to solve the joint data detection and parameter estimation problem. In the sequel, a joint estimation and detection algorithm is presented that can best be described in the framework of the iterative Gauss-Seidel scheme [7]. At one step, new estimates of $\underline{\tau}$ and $\underline{\alpha}$ are obtained while keeping $\hat{\mathcal{B}}$ constant. In a second iteration, $\mathcal{B}$ is detected using estimated values of $\hat{\underline{\tau}}$ and $\hat{\underline{\alpha}}$. The resulting algorithm is also suitable for practical scenarios where the signal parameters vary during data transmission and the detector can adapt to their changes.

3.1 Expectation Maximization Algorithm

In this section, we apply a powerful and practical algorithm to solve the optimization problem stated above. In particular, we use the EM-algorithm [8] to decompose the K-user maximum likelihood estimation problem into K simple and iterative optimization processes. The algorithm has been studied extensively and applied to a number of engineering problems.

In this application of the EM-algorithm, the received signal $r(t)$ is referred to as the incomplete data and the complete data is defined to be the decomposed signal $r(t)$, that is,

$$\underline{X}(t) = (X_1(t),\ldots,X_K(t))^T \text{ with } X_k(t) = S_k(t-\tau_k) + N_{t,k}, \text{ for } k = 1,\ldots,K, \quad (4)$$

where $N_t = \sum_{k=1}^{K} N_{t,k}$ and where all noise processes are chosen such that they are statistically independent. We first need to define Θ to be the set of all unknown parameters $(\underline{\tau},\underline{\alpha})$ and write the conditional log-likelihood function of the complete data as

$$U(\Theta,\Theta') \triangleq E\left[\log \mathcal{P}_{\underline{X}}(\underline{X};\Theta)|r(t),\Theta'\right]. \quad (5)$$

Let $\hat{\Theta}(m)$ denote the set of estimates available after the m^{th} step. The expectation maximization algorithm is built on the following result from statistics which states that

$$\begin{aligned} &\text{if} \quad \hat{\Theta}(m+1) = \arg\left\{\max_{\Theta}\left[U(\Theta,\hat{\Theta}(m))\right]\right\}, \\ &\text{then} \quad \Lambda(\hat{\Theta}(m+1)) \geq \Lambda(\hat{\Theta}(m)). \end{aligned} \quad (6)$$

This fact yields an iterative procedure that may solve the maximum likelihood problem. Starting with an initial estimate for the parameters denoted by $\hat{\Theta}(0)$, then at an instance of the algorithm, the E-step evaluates $U(\Theta, \hat{\Theta}(m))$ and the M-step maximizes $U(\Theta, \hat{\Theta}(m))$ with respect to Θ. The algorithm iterates till convergence or until a "satisfactory level of performance" is achieved.

Since the complete data is a vector of independent Gaussian random processes when conditioned on $r(t)$ and Θ, we can write

$$U(\Theta, \hat{\Theta}(m)) = c - \sum_{k=1}^{K} c_k \int |\hat{x}_k(t) - s_k(t - \tau_k)|^2 dt, \tag{7}$$

where c and the c_k are constants and $\hat{x}_k(t)$ is the conditional expectation of $X_k(t)$ given $r(t)$ and $\hat{\Theta}(m)$. The conditional expectation of $X_k(t)$ at the m^{th} step of the iteration is linear in $r(t)$ and can be written as [9]

$$\hat{x}_k(t) = \hat{s}_k(t - \hat{\tau}_k(m)) + \beta_k \left[r(t) - \sum_{k'=1}^{K} \hat{s}_{k'}(t - \hat{\tau}_{k'}(m)) \right], \tag{8}$$

where $\hat{s}_k(t)$ and $\hat{\tau}_k(m)$ are the estimates of the signal and the delay, respectively, that are available at the m^{th} step of the algorithm. The real coefficients β_k are, in principle, free parameters subject to the condition $\sum_{k=1}^{K} \beta_k = 1$ that can be adjusted for improved performance. The power of the algorithm lies in the fact that the maximization of the log-likelihood function over τ_k and α_k for $k = 1, \ldots, K$ has been decomposed into K independent and iterative procedures. Furthermore, every iteration guarantees no decrease in the value of the log-likelihood function.

We first consider estimating the delays $\tau_1, \ldots, \tau_K$. The maximization of the log-likelihood function of the complete data (7) can be simplified as

$$\hat{\tau}_k(m+1) = \arg\left\{ \max_{\tau_k} \left[\int \Re\{\hat{x}_k(t) \hat{s}_k^*(t - \tau_k)\} dt \right] \right\}, \quad \text{for } k = 1, \ldots, K. \tag{9}$$

Substituting (8) into (9) and manipulating the resulting equation yields the following maximization problem for the delay of the k^{th} signal:

$$\begin{aligned} \hat{\tau}_k(m+1) \;=\; & \arg\Bigg\{ \max_{\tau_k} \Bigg[|\hat{\alpha}_k(m)|^2 \sum_{i=-P}^{P} \sum_{i'=i-1}^{i+1} \hat{b}_k^{(i)}(m) \hat{b}_k^{(i')}(m) R_{k,k}(\hat{\tau}_k(m) - \tau_k + (i'-i)T) \\ & + \beta_k \sum_{i=-P}^{P} \hat{b}_k^{(i)}(m) \Bigg(\Re\left\{ \hat{\alpha}_k^*(m) z_k^{(i)}(\tau_k) \right\} \\ & - \sum_{i'=i-1}^{i+1} \sum_{k'=1}^{K} \hat{b}_{k'}^{(i')}(m) \, \Re\left\{ \hat{\alpha}_k^*(m) \hat{\alpha}_{k'}(m) \right\} R_{k',k}(\hat{\tau}_{k'}(m) - \tau_k + (i'-i)T) \Bigg) \Bigg] \Bigg\}, \end{aligned} \tag{10}$$

where $R_{k',k}(\tau) = \int a_{k'}(t - \tau) a_k(t) dt$ is the cross-correlation of the spreading waveforms $a_k(t)$ and $a_{k'}(t)$ and $z_k^{(i)}(\tau) = \int r(t) a_k(t - \tau - iT) dt$ is the output of the correlator with a candidate value of the delay.

In the next step within the $(m+1)^{st}$ iteration, the estimates of the α_k's will be updated. Since the log-likelihood is a quadratic function of the α_k's, the maximization of U yields a linear least-squares problem once estimates of the τ_k's and $b_k^{(i)}$'s are available.

Therefore, taking the derivative of $U(\Theta, \hat{\Theta}(m))$ with respect to α_k and setting it to zero provides an estimate of α_k as

$$\hat{\alpha}_k(m+1) = \frac{1}{(2P+1)T} \sum_{i=-P}^{P} \hat{b}_k^{(i)}(m) \int \hat{x}_k(t) a_k(t - \hat{\tau}_k(m) - iT)dt \quad \text{for } k = 1, \ldots, K. \quad (11)$$

Substituting (8) into (11) yields the following recursive estimator:

$$\begin{aligned} \hat{\alpha}_k(m+1) &= \hat{\alpha}_k(m) + \frac{\beta_k}{(2P+1)T} \sum_{i=-P}^{P} \hat{b}_k^{(i)}(m) \Bigg\{ z_k^{(i)}(\hat{\tau}_k(m)) \\ &- \sum_{i'=i-1}^{i+1} \sum_{k'=1}^{K} \hat{b}_{k'}^{(i')}(m) \hat{\alpha}_{k'}(m) R_{k',k}(\hat{\tau}_{k'}(m) - \hat{\tau}_k(m) + (i'-i)T) \Bigg\}, \quad (12) \end{aligned}$$

for each $k = 1, \ldots, K$.

The EM-algorithm applied to the parameter estimation problem for the CDMA system resulted in two sets of recursive equations, one for the delays, and one for the complex amplitudes. The algorithm can only guarantee convergence to a stable point of the likelihood function under certain regularity conditions. It is intuitive to expect convergence to a stationary point of the likelihood function that would result in a system with low average bit-error probabilities if the initial values of the parameters for the recursion are carefully chosen. Since there is always a sign ambiguity when considering joint data demodulation and carrier phase estimation in BPSK systems, we start transmission by sending a short preamble or a test pattern prior to data. The short preamble is used to establish initial estimates for the delays of the signals which can also be viewed as a rough timing acquisition. Considering a preamble of size $2P'+1$, the initial estimates for the delay and the complex amplitude of the k^{th} signal can be obtained from the position and the amplitude of the maximum of the k^{th} sliding correlator, that is,

$$\hat{\tau}_k(0) = \arg\left\{ \max_{\tau_k} \left| \sum_{i=-P'}^{P'} b_k^{(i)} z_k^{(i)}(\tau_k) \right| \right\}, \quad (13)$$

for the delay and

$$\hat{\alpha}_k(0) = \frac{1}{(2P'+1)T} \sum_{i=-P'}^{P'} b_k^{(i)} z_k^{(i)}(\hat{\tau}_k(0)), \quad (14)$$

for the complex amplitude of the signals for $k = 1, \ldots, K$.

3.2 Multistage Detection Algorithm

The multistage detection algorithm is a sub-optimum multiuser demodulation scheme that was first proposed by Varanasi and Aazhang in [4] for CDMA systems in white Gaussian channels. In this application of the multistage detector we will first consider a typical iteration of the maximum likelihood solution to detect the bits where the m^{th} estimates of the delays and the complex amplitudes are available, that is,

$$\hat{\mathcal{B}}(m+1) = \arg\{\max_{\mathcal{B}}[\Lambda(\mathcal{B}, \underline{\hat{\alpha}}(m), \underline{\hat{\tau}}(m))]\}. \quad (15)$$

The multistage algorithm finds the maximum likelihood decision on the i^{th} bit of the k^{th} user by iteratively solving the above equation while replacing the overlapping bits of the other users with their estimates from the previous stage. Therefore, the $(m+1)^{st}$ stage estimate of $b_k^{(i)}$ is written as

$$\hat{b}_k^{(i)}(m+1) = \arg\left\{ \max_{\substack{b_k^{(i)} \in \{-1,+1\} \\ b_{k'}^{(i')} = \hat{b}_{k'}^{(i')}(m)}} \left[\Lambda(\mathcal{B}, \underline{\hat{\alpha}}(m), \underline{\hat{\tau}}(m))\right] \right\}, \tag{16}$$

where the maximization is performed by setting $b_{k'}^{(i')}$ to be equal to its m^{th} stage estimate $\hat{b}_{k'}^{(i')}(m)$ for all $k' \neq k$ and $i' \neq i$. Considering the form of the log-likelihood function, the result of the maximization in (16) is given by

$$\begin{aligned} \hat{b}_k^{(i)}(m+1) \;=\; & \operatorname{sgn}\Bigg\{ \Re\left[\hat{\alpha}_k^*(m) z_k^{(i)}(\hat{\tau}_k(m))\right] - \sum_{k' \neq k} \Re\left[\hat{\alpha}_k^*(m)\hat{\alpha}_{k'}(m)\right] \times \\ & \Big(\hat{b}_{k'}^{(i-1)}(m) R_{k',k}(\hat{\tau}_{k'}(m) - \hat{\tau}_k(m) - T) + \hat{b}_{k'}^{(i)}(m) R_{k',k}(\hat{\tau}_{k'}(m) - \hat{\tau}_k(m)) \\ & + \hat{b}_{k'}^{(i+1)}(m) R_{k',k}(\hat{\tau}_{k'}(m) - \hat{\tau}_k(m) + T) \Big) \Bigg\} \end{aligned} \tag{17}$$

for all $k = 1, \ldots, K$ and $i = -P, \ldots, P$.

Using the initial estimates of the delays and the complex amplitudes as given by (13) and (14) after the preamble, the recursive procedure for data detection can be initialized according to the decision from the conventional matched filter which can be written as

$$\hat{b}_k^{(i)}(0) = \operatorname{sgn}\left\{ \Re\left[\hat{\alpha}_k^*(0) z_k^{(i)}(\hat{\tau}_k(0))\right] \right\}. \tag{18}$$

The effectiveness of the multistage detection algorithm in the context of joint detection and parameter estimation will be assessed in section 5.

4 Sequential Implementation

In view of an efficient implementation, the algorithm as proposed in section 3 has been modified to allow for sequential processing of the incoming signal. This is of practical importance since sequential algorithms may be superior to batch algorithms in terms of computational efficiency and memory requirements. Performing one iteration for the EM-algorithm per symbol period and denoting by $\hat{\tau}_k(i)$ and $\hat{\alpha}_k(i)$ the estimates from the i^{th} iteration, which corresponds to the i^{th} symbol interval, the updating rule for the sequential scheme is expressed as

$$\begin{aligned} \hat{\tau}_k(i) \;=\; & \arg\Bigg\{ \max_{\tau_k} \Bigg[|\hat{\alpha}_k(i-1)|^2 \sum_{i'=i-2P}^{i} \sum_{i''=i'-1}^{i'+1} \hat{b}_k^{(i')} \hat{b}_k^{(i'')} R_{k,k}(\hat{\tau}_k(i-1) - \tau_k + (i'' - i')T) \\ & + \beta_k \sum_{i'=i-2P}^{i} \hat{b}_k^{(i')} \Bigg(\Re\left\{\hat{\alpha}_k^*(i-1) z_k^{(i')}(\tau_k)\right\} \\ & - \sum_{i''=i'-1}^{i'+1} \sum_{k'=1}^{K} \hat{b}_{k'}^{(i'')} \Re\left\{\hat{\alpha}_k^*(i-1)\hat{\alpha}_{k'}(i-1)\right\} R_{k',k}(\hat{\tau}_{k'}(i-1) - \tau_k + (i''-i')T) \Bigg) \Bigg] \Bigg\} \end{aligned} \tag{19}$$

and

$$\begin{aligned}\hat{\alpha}_k(i) &= \hat{\alpha}_k(i-1) + \frac{\beta_k}{(2P+1)T} \sum_{i'=i-2P}^{i} \hat{b}_k^{(i')} \Bigg\{ z_k^{(i')}(\hat{\tau}_k(i-1)) \\ &\quad - \sum_{i''=i'-1}^{i'+1} \sum_{k'=1}^{K} \hat{b}_{k'}^{(i'')} \hat{\alpha}_{k'}(i-1) R_{k',k}(\hat{\tau}_{k'}(i-1) - \hat{\tau}_k(i-1) + (i''-i')T) \Bigg\}. \qquad (20)\end{aligned}$$

This "sliding window"-algorithm therefore obtains the i^{th} parameter estimates by processing the last available $2P+1$ symbols at time i. In both (19) and (20), $\hat{b}_k^{(i)}$ stands for the "best" available estmate of $b_k^{(i)}$. It has been shown in previous work [4] that two stages are sufficient to eliminate the near-far effect. With the above values $\hat{\tau}_k(i)$ and $\hat{\alpha}_k(i)$, the estimates for $b_k^{(i-1)}$ and $b_k^{(i-2)}$ are in turn updated according to (17) in order to obtain $\hat{b}_k^{(i-1)}(1)$ and $\hat{b}_k^{(i-2)}(2)$ for all $k = 1 \ldots K$.

5 Numerical Results and Conclusions

Since it may be very difficult or even impossible to find analytical expressions for the performance of the proposed receiver structure, we restrict the analysis in this work to Monte Carlo simulations. The results given in this section are based on simulations of a two user-system ($K = 2$) using the first two m-sequences of length $N_c = 31$ as listed in [10]. As the proposed interference rejection scheme is feasible only by means of an all-digital implementation, the received signal $r(t)$ is sampled and processed at discrete time instants spaced $\Delta t = T_c/N_s$ time units apart. In all examples, the number of samples per chip was selected to be $N_s = 4$, with an observation span of $(2P+1) = 7$ symbols.

It has only been assumed that the receiver knows for all users $k = 1 \ldots K$ the index i_k of the time interval $[\, i_k T, (i_k+1)T\,)$ where the k^{th} user starts transmission of a packet. However, the delays τ_k of the individual users relative to the symbol interval boundaries can assume any value $\tau_k \in [0, T)$. Furthermore, in order to overcome the two-fold phase ambiguity of antipodal signaling, a short preamble of length $(2P'+1)$ has been added to each packet to help the receiver obtain amplitude estimates with the correct polarity. It has been found that a value as small as $P' = 3$ is sufficient for the initialization of the algorithm according to (13) and (14).

In Figure 1, the bit error probability $P_{b,1}$ of user 1 is plotted versus its signal-to-noise ratio $\gamma_1 = E_1/\mathcal{N}_0$ for a two-user system ($K = 2$) with user 2 being a strong interferer ($E_2/E_1 = 8\,\mathrm{dB}$). When comparing the proposed iterative scheme to the conventional matched filter detector which even operates with perfect knowledge of the code delay τ_1 and the complex-valued amplitude α_1, it is seen that a substantial improvement in performance can be achieved which is very close to the bit error probability of a single-user detector, as denoted by the dotted line. Figure 2 shows the sensitivity of the multiuser detection and estimation algorithm with respect to the interference level. Again, the bit error probability $P_{b,1}$ is depicted, but keeping γ_1 at a constant value of $8\,\mathrm{dB}$ while varying the symbol energy E_2 of the interfering user relative to E_1. It can be seen from this figure that the proposed algorithm is very insensitive to the relative interference level, and therefore is able to overcome the interference-limited behaviour of conventional CDMA-systems.

In order to determine the required length of the preamble which is employed to overcome the problem of phase-ambiguity, it is instructive to examine the transient behaviour when estimating the channel parameters α_k and τ_k. Figures 3 and 4 depict the root-mean-square error of the amplitude and the delay estimates $\hat{\alpha}_1(i)$ and $\hat{\tau}_1(i)$ respectively as a function of the iteration step i which corresponds to the number of processed symbols in the sequential implementation of the scheme. It can be observed that the delay estimate can accurately be obtained upon receipt of the fifth symbol, whereas the amplitude estimation error reaches the steady state after approximately 10 symbols. Consequently, a preamble of $2P' + 1 = 7$ symbols has been found to be sufficiently long in order to obtain reliable estimates of the channel parameters. Since data detection can only take place after the preamble, we conclude that Figs. 1 and 2 illustrate the steady-state error performance of the multiuser detection and estimation scheme, i.e. the bit error rates are very close to those of a single-user receiver with perfect knowledge of the delay and carrier phase due to the high accuracy of the parameter estimates obtained from processing the preamble. It can also be seen from Figs. 3 and 4 that the parameter estimation is very insensitive to variations of E_2/E_1 after the second iteration.

Further studies will include transmission over slowly fading channels in order to investigate the dynamic behaviour of the parameter estimation scheme. Moreover, the delay estimation is basically a search procedure since the signal is processed at discrete time instants. The computational complexity of the τ-search could considerably be reduced by applying combinatorial optimization schemes, e.g., branch-and-bound techniques. Such local search methods can play an important role when implementing multiuser receivers.

References

[1] S. Verdu, “Minimum probability of error for asynchronous Gaussian multiple-access channels,” *IEEE Trans. Inform. Theory*, vol. IT-32, pp. 85–96, Jan. 1986.

[2] H. V. Poor and S. Verdu, “Single-user detectors for multiuser channels,” *IEEE Trans. Commun.*, vol. COM-36, pp. 50–60, Jan. 1988.

[3] R. Lupas and S. Verdu, “Linear multiuser detectors for synchronous code-division multiple-access channels,” *IEEE Trans. Inform. Theory*, vol. IT-35, pp. 123–136, Jan. 1989.

[4] M. K. Varanasi and B. Aazhang, “Multistage detection in asynchronous code-division multiple-access communications,” *IEEE Trans. Commun.*, vol. COM-38, pp. 509–519, Apr. 1990.

[5] Z. Xie, C. R. Rushforth, and R. T. Short, “Multiuser signal detection using sequential decoding,” *IEEE Trans. Commun.*, vol. 38, pp. 578–583, May 1990.

[6] B. Aazhang, B.-P. Paris, and G. Orsak, “Neural networks for multiuser detection in code-division multiple-access communications,” *IEEE Trans. Commun.*, vol. COM-40, pp. 1212–1222, July 1992.

[7] J. M. Ortega and W. C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables.* New York, NY: Academic Press, 1970.

[8] A. P. Dempster, N. M. Laird, and D. B. Rubin, “Maximum likelihood from incomplete data via the EM algorithm,” *J. Royal Statist. Soc., Ser. B*, vol. 39, no. 1, pp. 1–38, 1977.

[9] M. Feder and E. Weinstein, “Parameter estimation of superimposed signals using the EM algorithm,” *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-36, pp. 477–489, Apr. 1988.

[10] M. B. Pursley and H. F. A. Roefs, “Numerical evaluation of correlation parameters for optimal phases of binary shift-register sequences,” *IEEE Trans. Commun.*, vol. COM-27, pp. 1597–1604, Oct. 1979.

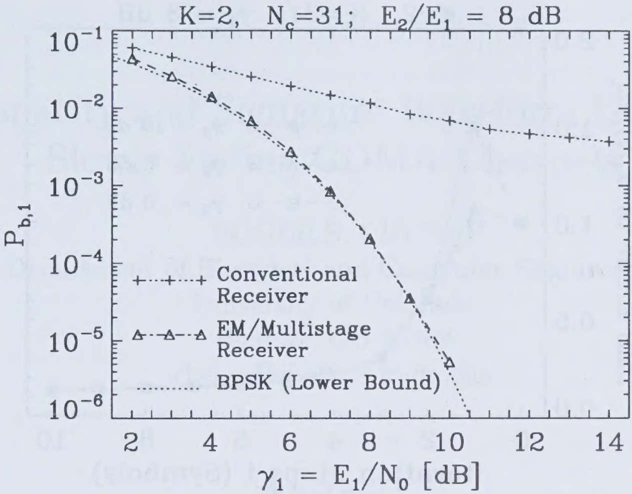


Figure 1: Bit error probability of user 1 as a function of the signal-to-noise ratio $\gamma_1 = E_1/\mathcal{N}_0$ in the presence of a second user with $E_2/E_1 = 8\,\mathrm{dB}$.

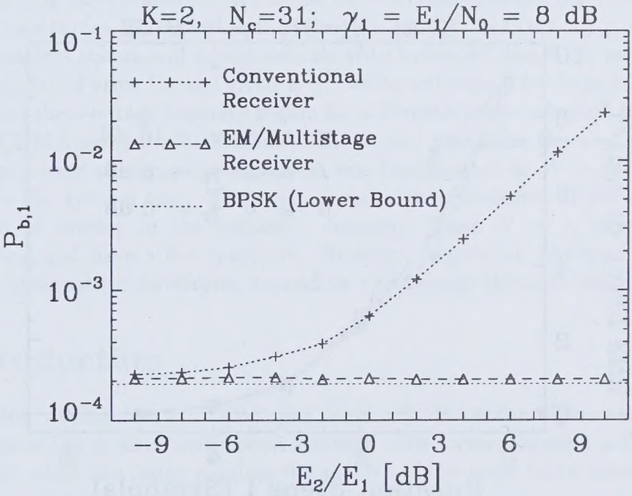


Figure 2: Bit error probability of user 1 (at a constant signal-to-noise ratio $\gamma_1 = 8\,\mathrm{dB}$) as a function of the multiple access interference level expressed as the ratio of symbol energies E_2/E_1.

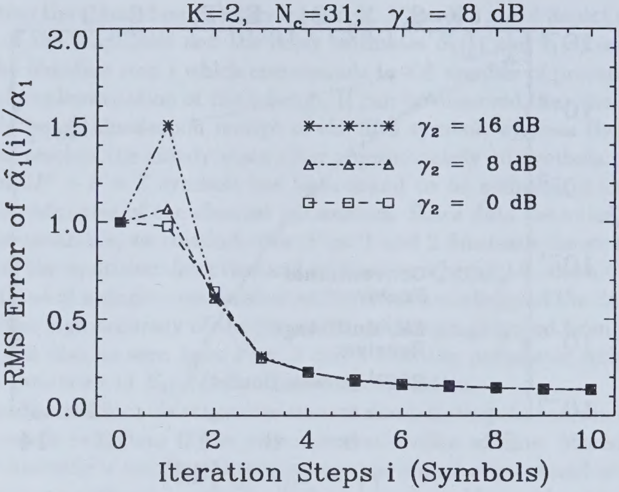


Figure 3: Normalized root-mean-square error of the amplitude estimate of user 1 in the presence of an interfering signal.

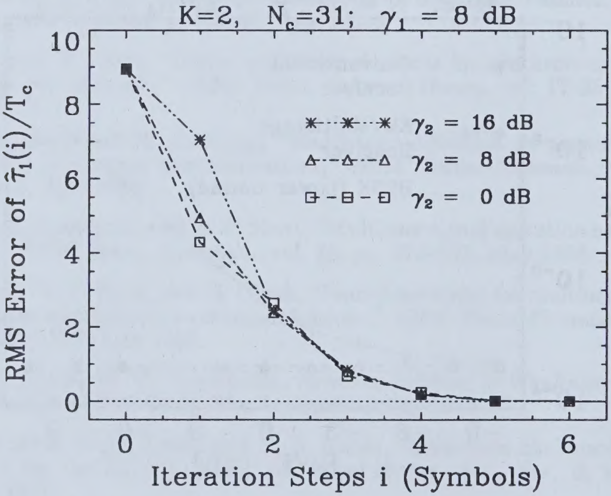


Figure 4: Normalized root-mean-square error of the delay estimate of user 1 in the presence of an interfering signal.

On Capacity and Signature Waveform Design for Slowly Fading CDMA Channels

ROGER S. CHENG
Department of Electrical and Computer Engineering
University of Colorado
Boulder, CO 80309
chengr@spot.colorado.edu

Abstract

We model the slowly fading code-division multiaccess (CDMA) channel by an non-ergodic channel where the fading parameters remain constant for all time once they are chosen randomly. We assume that the receiver knows the actual realizations of the fading parameters, **a**, while the transmitter knows only their statistics. We define the ϵ-outage capacity region as the set of all rate tuples, $\mathbf{R}$, such that there exists a set S with $P(\mathbf{a} \in S) \geq 1-\epsilon$, and independent coders and a joint decoder with codeword length N, rate $\mathbf{R}$, and probability of error for any given $\mathbf{a} \in S$ arbitrarily small for large enough N. We find the ϵ-outage capacity region for a frequency non-selective fading K-user CDMA symbol-synchronous channel, and maximize the ϵ-outage total capacity over all signature waveform sets bandlimited to $W = N/T$ where $1/T$ is the symbol rate. The optimal signature waveforms of the users are shown to overlap in the frequency domain. When $N = 1$, they are are identical and have a flat spectrum. However, in general, the spectra of the optimal signature waveforms depend on the average signal-to-noise ratios.

I Introduction

Code division multiaccess (CDMA) and frequency division multiaccess (FDMA) are special cases of the general multiaccess strategy. The former imposes a structure in the time domain while the latter requires the spectra of the users to be non-overlapping in the frequency domain.

It is known that [1] FDMA is optimal in achieving the total capacity of a waveform channel with additive Gaussian noise. However, it is not clear whether CDMA can also achieve the total capacity. When the users are synchronized and have equal power, a necessary and sufficient condition on the signature waveforms for CDMA to achieve

---

This work was supported by the National Science Foundation under Grant NCR-92-9812

the total capacity was obtained in [2], and it can be shown that there always exists signature waveform set satisfying that condition. Hence, if the signature waveform set is optimally designed, CDMA can also achieve the total capacity for the symbol-synchronous channel with equal user power. In Section II, we obtain the capacity region of a symbol-asynchronous CDMA channel as a function of the signature waveform set. We also optimize the capacity region over all bandlimited signature waveform set for fixed spreading factor N. Then, we find the necessary and sufficient condition on the users' powers such that there exists signature waveforms set for CDMA to achieve the total capacity of the waveform Gaussian noise channel.

While FDMA is optimal in achieving the total capacity in Gaussian channels and under a wide range of users' powers, CDMA is also optimal with properly designed signature waveforms, the situation is entirely different for fading channels. The capacity and the capacity region of frequency non-selective fading single-user and multiuser channels were found in terms of the Exponential-Integral function in [3, 4]. It shows that the total capacity achieved by FDMA is strictly less than that by general multiaccess strategy, though the difference is rather small.

These are asymptotic results where the codeword length can be arbitrarily long. In particular, the results only reflect the situation where the codeword length is long enough for us to consider only the average behavior of the fading process. In some channels with slow fading, this assumption is not valid and the capacity result does not provide a performance limit on the channel.

In Section III, we consider the other extreme where we assume that the fading process remains constant over the duration of the transmission of a codeword. Hence, the maximum rate of reliably transmission depends on the realization of the fading process. We define the the ϵ-outage capacity region as the collection of all rate tuples such that there exists a set Ω with probability larger than $1-\epsilon$, and encoders and decoders with probability of error asymptotically small for every fading process realization in Ω. We also define the outage probability for any rate tuple $\mathbf{R}$ as the minimum ϵ such that $\mathbf{R}$ is in the ϵ-outage capacity region. We find that in contrast to the Gaussian noise channel, for a given outage probability, the rate tuple achieved by CDMA with properly designed signature waveform set can be significantly larger than that by FDMA in a frequency non-selective fading channels.

II Capacity Regions for Various Multiaccess Schemes

In this section, we consider three different multiaccess schemes on a strictly bandlimited passband multiaccess channel with bandwidth B. The equivalent baseband channel is strictly bandlimited to $[-B/2, B/2]$ and the received signal is given by

$$\begin{aligned} Y^{(I)}(t) &= \sum_{k=1}^{K} X_k^{(I)}(t) * H_k(t-\tau_k) + N^{(I)}(t), \\ Y^{(Q)}(t) &= \sum_{k=1}^{K} X_k^{(Q)}(t) * H_k(t-\tau_k) + N^{(Q)}(t), \end{aligned} \qquad (1)$$

where the superscript I and Q stand for the in-phase and the quadrature-phase components, respectively, and $*$ denotes the convolution operation. In (1), $H_k(t)$ is the complex impulse response of the k^{th} user equivalent baseband channel, and $X_k^{(I)}(t)$ and $X_k^{(Q)}(t)$ are the signals sent by the k^{th} user in the I and Q channels, respectively. $N^{(I)}(t)$ and $N^{(Q)}(t)$ are independent white Gaussian noise processes, each with power spectral density level σ^2. We assume that the users' signals, after passing through respective channels, are superimposed asynchronously and τ_k is the relative delay of the k^{th} user.

The first multiaccess scheme we considered is the most general one in which the codewords of the users have to be strictly bandlimited and satisfy the power constraint, i.e.

$$\frac{1}{T}\int_0^T (X_k^{(I)}(t))^2 + (X_k^{(Q)}(t))^2 dt \leq w_k, \qquad k = 1, \dots, K, \tag{2}$$

where w_k is the transmit power of the k^{th} user in unit energy per second. We will refer to the channel with this general multiaccess scheme the GMAC.

The second multiaccess scheme we considered is the FDMA scheme in which, in addition to the bandwidth and the power constraints, the input power spectral densities of the users must be non-overlapping in the frequency domain.

The third multiaccess scheme is the CDMA strategy (in a restricted sense) in which, in addition to the bandwidth and the power constraint, the codewords of the k^{th} user must have the form

$$\begin{aligned} X_k^{(I)}(t) &= \sum_j b_{kj}^{(I)} s_k^{(I)}(t - jN/B), \\ X_k^{(Q)}(t) &= \sum_j b_{kj}^{(Q)} s_k^{(Q)}(t - jN/B), \end{aligned} \tag{3}$$

where $s_k(t)$ is the unit-energy signature waveform for the k^{th} user, and N is the spreading factor. Notice that this is a special case of the general spread spectrum CDMA scheme because we have assumed that all information symbols of the k^{th} user, b_{kj}, are modulated by the same waveform. The corresponds to the special case where the length of the pseudorandom sequence is equal to the spreading factor, N.

Since FDMA and CDMA are special cases of the GMAC strategy, the capacity regions of the FDMA and the CDMA channels must be subsets of the capacity region of the GMAC.

The capacity region of the FDMA channel follows straightforwardly from the results in single-user channels because the non-overlapping input power spectral densities ensure zero interuser interference, regardless of the relative delays between the users. The capacity region is

$$C_{\text{fdma}}(B, \mathbf{w}) = \bigcup_{\Omega_k, \Sigma_k(w), k=1,\dots,K} \left\{(R_1, \dots, R_K) : R_k \leq \frac{B}{2\pi}\int_{\Omega_k} \log[1 + |H_k(Bw)|^2 \Sigma_k(w)]dw\right\}, \tag{4}$$

where the union is taken over all disjoint $\Omega_k \in [-\pi, \pi]$ and $\Sigma_k(w)$ defined on Ω_k with

$$\frac{1}{2\pi}\int_{\Omega_k} \Sigma_k(w)dw = \frac{w_k}{2B\sigma^2},$$

for $k = 1, \dots, K$, and $H_k(w)$ is the transfer function of the channel seen by the k^{th} user.

Using the result in [1], we find the capacity region of the CDMA channel in the following theorem. (This result also generalizes the capacity region found in [5] to the case with inter-symbol interference.)

Theorem 1

The capacity region of the asynchronous CDMA channel for any signature waveform set, $\mathbf{s}(t)$, is given by

$$\mathrm{C}_{\mathrm{cdma}}(\mathbf{s}(t), B, \mathbf{w}) = \bigcup_{\Sigma_k(w), k=1,\ldots,K} \; \bigcap_{\tau_k, k=1,\ldots,K}$$

$$\left\{ \begin{array}{l} (R_1, \ldots, R_K) : \forall \mathcal{J} \subseteq \{1, \ldots, K\} \\ \sum_{j \in \mathcal{J}} R_j \leq \frac{B}{2N\pi} \int_{-\pi}^{\pi} \log\det[\mathbf{I} + \sum_{k \in \mathcal{J}} D_k(w) S_k(w) \Sigma_k(w) S_k^*(w) D_k^*(w)]dw \end{array} \right\}, \quad (5)$$

where the union is taken over all $\Sigma_k(w) \in \mathbb{R}^{2\times 2}$ such that

$$\frac{1}{2\pi}\int_{-\pi}^{\pi} \mathrm{tr}(S_k(w)\Sigma_k(w)S_k^*(w))dw = \frac{Nw_k}{2B\sigma^2},$$

for $k = 1, \ldots, K$. In (5),

$$[D_k(w)]_{mn} = \left\{ \begin{array}{ll} H_k(\frac{B(w-(N-2n+1)\pi)}{N}) e^{j\frac{B(N-2n+1)\pi}{N}\tau_k} & \text{if } m = n, \\ 0 & \text{otherwise,} \end{array} \right.$$

and $S_k(w) = [S_k^{(I)}(w) S_k^{(Q)}(w)]$ where

$$[S_k^{(p)}(w)]_n = S_k^{(p)}\left(\frac{B(w-(N-2n+1)\pi)}{N}\right), \qquad p = I, Q, \quad n = 1, \ldots, N,$$

are the Nyquist translates of the frequency spectra of the signature waveforms in the I and Q channels, respectively. □

Notice that the capacity region depends on the signature waveforms and the input power spectral density of the data b_{kj} through their product $S_k(w)\Sigma_k(w)S_k^*(w)$. Hence, if we further optimize over the signature waveform set, we have the following theorem.

Theorem 2

The capacity region of the asynchronous CDMA channel (with optimal signature waveform set) is given by

$$\mathrm{C}_{\mathrm{cdma}}(B, \mathbf{w}) = \bigcup_{S_k(w), k=1,\ldots,K} \; \bigcap_{\tau_k, k=1,\ldots,K}$$

$$\left\{ \begin{array}{l} (R_1, \ldots, R_K) : \forall \mathcal{J} \subseteq \{1, \ldots, K\} \\ \sum_{j \in \mathcal{J}} R_j \leq \frac{B}{2N\pi} \int_{-\pi}^{\pi} \log\det[\mathbf{I} + \sum_{k \in \mathcal{J}} D_k(w) S_k(w) S_k^*(w) D_k^*(w)]dw \end{array} \right\}, \quad (6)$$

where the union is taken over all $S_k(w) \in \mathbb{R}^{N\times 2}$ such that

$$\frac{1}{2\pi}\int_{-\pi}^{\pi} \mathrm{tr}(S_k(w)S_k^*(w))dw = \frac{Nw_k}{2B\sigma^2},$$

for $k = 1, \ldots, K$, and

$$[D_k(w)]_{mn} = \left\{ \begin{array}{ll} H_k(\frac{B(w-(N-2n+1)\pi)}{N}) e^{j\frac{B(N-2n+1)\pi}{N}\tau_k} & \text{if } m = n, \\ 0 & \text{otherwise.} \end{array} \right.$$

□

In Theorem 2, we have assumed, without loss of generality, that the power spectral density of the data b_{kj} is constant and the input power spectral density to the channel is shaped by the signature waveform set. Hence, the optimal $S_k(w) = [S_k^{(I)}(w)S_k^{(Q)}(w)]$ in (6) gives the optimal signature waveform set.

In fact, the capacity region in Theorem 2 can be viewed as the general capacity region formula. If N=1 and the union in (6) is taken over all $S_k(w)$, $k = 1, \ldots, K$ with disjoint supports, we have the capacity formula for the FDMA channel. On the other hand, the GMAC is the same as the CDMA channel with $N = 1$ and $s_k(t) = \sqrt{B}\mathrm{sinc}(Bt)$. Hence, we have the following corollary for the GMAC.

Corollary 1

The capacity region of the asynchronous GMAC is given by

$$C_{\mathrm{gmac}}(B, \mathbf{w}) =$$

$$\bigcup_{\Sigma_k(w), k=1,\ldots,K} \left\{ \begin{array}{l} (R_1, \ldots, R_K) : \forall \mathcal{J} \subseteq \{1, \ldots, K\} \\ \sum_{j\in\mathcal{J}} R_j \leq \frac{B}{2\pi} \int_{-\pi}^{\pi} \log[1 + \sum_{k\in\mathcal{J}} |H_k(Bw)|^2 \Sigma_k(w)]dw \end{array} \right\}, \tag{7}$$

where the union is taken over all $\Sigma_k(w)$ such that

$$\frac{1}{2\pi} \int_{-\pi}^{\pi} \Sigma_k(w)dw = \frac{w_k}{2B\sigma^2},$$

for $k = 1, \ldots, K$. □

Corollary 1 is a special case of the results in [1] where we have shown that the total capacity of the GMAC, C_{gmac}, is achieved by FDMA signaling. Therefore, we have that $C_{\mathrm{gmac}}(B, \mathbf{w}) = C_{\mathrm{fdma}}(B, \mathbf{w})$. Moreover, since the CDMA channel is the GMAC with additional restriction on the codewords, we have that $C_{\mathrm{cdma}}(B, \mathbf{w}) \leq C_{\mathrm{gmac}}(B, \mathbf{w}) = C_{\mathrm{fdma}}(B, \mathbf{w})$. The following corollary gives a necessary and sufficient condition on the users' powers for $C_{\mathrm{cdma}}(B, \mathbf{w}) = C_{\mathrm{gmac}}(B, \mathbf{w}) = C_{\mathrm{fdma}}(B, \mathbf{w})$.

Corollary 2

Suppose that $H_k(w) = h_k$ for $k = 1, \ldots, K$, and let $C_{\mathrm{cdma}}(B, \mathbf{w})$, $C_{\mathrm{fdma}}(B, \mathbf{w})$, and $C_{\mathrm{gmac}}(B, \mathbf{w})$ be the total capacities of the asynchronous CDMA, FDMA, and GMAC channels, respectively. Then, we have

$$C_{\mathrm{cdma}}(B, \mathbf{w}) \leq C_{\mathrm{gmac}}(B, \mathbf{w}) = C_{\mathrm{fdma}}(B, \mathbf{w}),$$

with equality iff $|h_k|^2 w_k / (\sum_{k=1}^{K} |h_k|^2 w_k) \leq 2/N$, $\forall k = 1, \ldots, K$. □

From this corollary, we see that the spreading factor in a CDMA AWGN channel should always be less than twice the number of users in order for CDMA to achieve the total capacity achieved by FDMA signaling.

III ϵ-Outage Capacity Region for Fading Channels

In this section, we will use the result in the previous section to study the ϵ-outage capacity region of fading CDMA, FDMA, and GMAC channels. We will begin with the definition of the ϵ-outage capacity region in Section III.1. Then, we study the frequency non-selective fading channel in Section III.2.

III.1 Definition of ϵ-Outage Capacity Region

A fading channel is a linear time-varying channel with complex impulse response $h(\tau;t)$ which is a random process in two parameters. The output of the channel with input $u(t)$ is

$$y(t) = \int u(t-\tau)h(\tau;t)d\tau + n(t).$$

If $h(\tau;t) = c(t)\delta(\tau)$ for all t, the channel is referred to as the frequency non-selective fading channel because of its flat instantaneous frequency response. Otherwise, the channel is referred to as the frequency selective fading channel.

If we assume that for each τ, $h(\tau;t)$ is a stationary ergodic random process in the parameter t and that the receiver has complete knowledge of the fading process $h(\tau;t)$, the capacity of the fading channel can be characterized in a limiting expression [6]. However, close form expressions for the capacity and the capacity region of the single-user and multiuser fading channels have not been found except in special cases. In particular, if $n(t)$ is white Gaussian noise and the channel is frequency non-selective fading, the capacity is the expectation of the capacity as a function of the fading parameter $c(t)$. If we further assume that $c(t)$ is a stationary complex Gaussian process, the capacity and the capacity region can be expressed using the Exponential-Integral function [3, 4].

As in all information theoretic results, these capacity formulas for fading channels are asymptotic results with arbitrarily long codeword length. In particular, these results are only valid if the codeword length is long enough such that the effect due to fading is averaged out. However, in most practical situations, $h(\tau;t)$ varies slowly with t and it is unreasonable to assume that the codeword length is long enough to consider only the average behavior of the fading process.

In this paper, we will consider the other extreme where we assume that the fading process does not vary with time during the duration of transmitting a codeword. Under this assumption, $h(\tau;t) = h(\tau)$ and the channel is non-ergodic. It had been pointed out in [3] that the capacity is zero since the fading process can be arbitrarily close to zero with non-zero probability. This fact motivates us to define the ϵ-outage capacity region in the following way.

Definition: A K-tuple $\mathbf{R}$ is ϵ-outage achievable for a K-user multiaccess fading channel if there exists a set, $\Omega \in \mathbf{C}^K(\mathrm{I\!R})$, with $P((h_1(\tau),\ldots,h_k(\tau)) \in \Omega) \geq 1-\epsilon$, and a series of length N independent coders and joint decoders with rate $\mathbf{R}$, such that the probability of error for any given channel realizations $(h_1(\tau),\ldots,h_k(\tau)) \in \Omega$ is arbitrarily small for large enough N.

Definition: The ϵ-outage capacity region, $\mathbf{C}(\epsilon)$, of a K-user multiaccess fading channel is defined as the set of all ϵ-outage achievable K-tuples $\mathbf{R}$.

Definition: The outage probability of a K-tuple, $P_O(\mathbf{R})$ is defined as

$$P_O(\mathbf{R}) = \min_{\mathbf{C}(\epsilon):\mathbf{R}\in\mathbf{C}(\epsilon)} \epsilon$$

It is important to note that the encoders and the decoder are not functions of the fading process which is a consequence of our assumption that neither the transmitter nor the receiver know the fading process.

The ϵ-outage capacity region defined above is different from the ϵ-capacity (e.g. in [7]) although the two definitions coincide when the strong converse holds for the underlying channel with every realization of the fading process.

III.2 Frequency Non-Selective Fading Channels

We consider the frequency non-selective fading channel where the fading process $h(\tau) = \alpha\delta(\tau)$. The received signal of this channel is

$$
\begin{aligned}
Y^{(I)}(t) &= \sum_{k=1}^{K} \alpha_k X_k^{(I)}(t-\tau_k) + N^{(I)}(t), \\
Y^{(Q)}(t) &= \sum_{k=1}^{K} \alpha_k X_k^{(Q)}(t-\tau_k) + N^{(Q)}(t).
\end{aligned} \tag{8}
$$

Applying the result in compound channel (e.g. [7]) and the definition of the ϵ-outage capacity region, we have the following theorem.

Theorem 3

The ϵ-outage capacity region of the asynchronous CDMA frequency non-selective fading channel in (8) is given by

$$
\mathrm{C}^{f}_{\mathrm{cdma}}(\epsilon) = \bigcup_{S_k(w),k=1,\ldots,K}
\left\{ \begin{array}{l} (R_1,\ldots,R_K): \\ P\left(\begin{array}{l} \forall \mathcal{J} \subseteq \mathcal{K}, \sum_{j\in\mathcal{J}} R_j \le \\ \min\limits_{\tau_k,k\in\mathcal{J}} \dfrac{B}{2N\pi} \displaystyle\int_{-\pi}^{\pi} \log\det[\mathbf{I} + \sum_{k\in\mathcal{J}} |\alpha_k|^2 \Xi_k(w) S_k(w) S_k^T(w) \Xi_k^*(w)]dw \end{array} \right) \\ \ge 1-\epsilon, \end{array} \right\}, \tag{9}
$$

where the union is taken over all $S_k(w) \in \mathbb{R}^{N\times 2}$ such that

$$
\frac{1}{2\pi}\int_{-\pi}^{\pi} \mathrm{tr}(S_k(w)S_k^T(w))dw = \frac{Nw_k}{2B\sigma^2},
$$

for $k = 1, \ldots, K$, and

$$
[\Xi_k(w)]_{mn} = \begin{cases} e^{j\frac{B(N-2n+1)\pi}{N}\tau_k} & \text{if } m = n, \\ 0 & \text{otherwise.} \end{cases}
$$

The outage probability for any rate tuple $\mathbf{R}$ is

$$
P_{o,\mathrm{cdma},N}(\mathbf{R}) = 1 - \sup_{S_k(w),k=1,\ldots,K}
P\left(\begin{array}{l} \forall \mathcal{J} \subseteq \mathcal{K}, \sum_{j\in\mathcal{J}} R_j \le \\ \min\limits_{\tau_k,k\in\mathcal{J}} \dfrac{B}{2N\pi} \displaystyle\int_{-\pi}^{\pi} \log\det[\mathbf{I} + \sum_{k\in\mathcal{J}} |\alpha_k|^2 \Xi_k(w) S_k(w) S_k^T(w) \Xi_k^*(w)]dw \end{array} \right). \tag{10}
$$

□

Corollary 3

The ϵ-outage capacity region of the asynchronous frequency non-selective fading GMAC in (8) is given by

$$\mathrm{C}^f_{\mathrm{gmac}}(\epsilon) = \left\{(R_1, \ldots, R_K) : P\left(\forall \mathcal{J} \subseteq \mathcal{K}, \sum_{j\in\mathcal{J}} \frac{|\alpha_j|^2 w_k}{2B\sigma^2} \geq \exp(\frac{1}{B}\sum_{j\in\mathcal{J}} R_j) - 1\right) \geq 1-\epsilon\right\}.$$

The outage probability for any rate tuple $\mathbf{R}$ is

$$P_{O,\mathrm{gmac}}(\mathbf{R}) = 1 - P\left(\forall \mathcal{J} \subseteq \mathcal{K}, \sum_{j\in\mathcal{J}} \frac{|\alpha_j|^2 w_j}{2B\sigma^2} \geq \exp(\frac{1}{B}\sum_{j\in\mathcal{J}} R_j) - 1\right). \tag{11}$$

The optimal input power spectral densities of all the users are identical and is a constant over the entire allocation frequency band with width B. □

Corollary 4

The ϵ-outage capacity region of the asynchronous frequency non-selective fading FDMA channel in (8) is given by

$$\mathrm{C}^f_{\mathrm{fdma}}(\epsilon) = \left\{(R_1, \ldots, R_K) : \max_{\boldsymbol{\beta}:\sum_{k=1}^K \beta_k \leq 1} P\left(\forall k \in \mathcal{K}, \frac{|\alpha_k|^2 w_k}{2\beta_k B\sigma^2} \geq \exp(\frac{R_k}{\beta_k B}) - 1\right) \geq 1-\epsilon\right\}. \tag{12}$$

The outage probability for any rate tuple $\mathbf{R}$ is

$$P_{O,\mathrm{fdma}}(\mathbf{R}) = 1 - \max_{\boldsymbol{\beta}:\sum_{k=1}^K \beta_k \leq 1} P\left(\forall k \in \mathcal{K}, \frac{|\alpha_k|^2 w_k}{2\beta_k B\sigma^2} \geq \exp(\frac{R_k}{\beta_k B}) - 1\right). \tag{13}$$

The optimal input power spectral density of the k^{th} user is a constant over its optimal frequency band with width $\beta_k B$. □

It is clear from the definition of the various multiaccess strategies that $\mathrm{C}^f_{\mathrm{cdma}}(\epsilon) \subseteq \mathrm{C}^f_{\mathrm{gmac}}(\epsilon)$ and $\mathrm{C}^f_{\mathrm{fdma}}(\epsilon) \subseteq \mathrm{C}^f_{\mathrm{gmac}}(\epsilon)$. However, unlike the Gaussian channel where FDMA is optimal for achieving the total capacity, we have that $C^f_{\mathrm{fdma}}(\epsilon) < C^f_{\mathrm{gmac}}(\epsilon)$ for all ϵ if $\{\alpha_k\}_{k=1}^K$ are independent. Hence, FDMA is no longer optimal for frequency non-selective fading channels. This can be explained by the fact that the variation in the users' powers due to fading actually helps the receiver to distinguish the users' signals.

In Figures 1, 2, and 3, we plotted that outage probabilities of the GMAC and FDMA frequency non-selective fading channels versus R/B for $K = 2$, 5, and 10, respectively. We note that for a given outage probability, say 10%, the general multiaccess strategy can achieve a much higher rate, sometimes more than twice the rate achieved by FDMA. Moreover, the difference becomes larger as K increases.

It is of interest to compare the performance achieved by FDMA and CDMA. While the two schemes yield the same total capacity in the ideal all-pass AWGN channel when $|h_k|^2 w_k / (\sum_{k=1}^K |h_k|^2 w_k) \leq 2/N$, $\forall k = 1, \ldots, K$, this is no longer the case for frequency non-selective fading channel, even though every realization of the fading process still gives a flat transfer function. We assume independent Rayleigh fading where $\{\alpha_k\}_{k=1}^K$ are independent zero-mean complex Gaussian random variables with $\mathrm{Var}(\mathrm{Re}(\alpha_k)) = \mathrm{Var}(\mathrm{Im}(\alpha_k))$. The following corollary gives an upper bound to the outage probability of the rate tuple $\mathbf{R} = R\mathbf{1}$ for the CDMA channel and its relationship with that of FDMA channel.

Corollary 5

Suppose that $\{\alpha_k\}_{k=1}^K$ are independent zero-mean complex Gaussian random variables and $\mathbf{E}\frac{|\alpha_k|^2 w_k}{2B\sigma^2} = \lambda$ for all $k = 1, \ldots, K$. Then, we have that $P_{o,\mathrm{cdma},N}(R\mathbf{1})$ is upper bounded by

$$\bar{P}_{o,\mathrm{cdma},N}(R\mathbf{1}) \triangleq 1 -$$

$$P\left(\forall \mathcal{J} \in \mathcal{K}, |\mathcal{J}|R \leq \frac{B}{N}\sum_{n=0}^{N-1} \log\left[1 + N\left(\sum_{\substack{j\in\mathcal{J}: \\ (j)_N = n}} \frac{r_k}{2} + \sum_{\substack{j\in\mathcal{J}: \\ (j+1)_N = n}} \frac{r_k}{2}\right)\right]\right), \quad (14)$$

where $(j)_N$ denotes j mod N and $r_k = \frac{|\alpha_k|^2 w_k}{2B\sigma^2}$. The upper bound is achieved by signature waveforms corresponding to $S_k(w) = \frac{N w_k}{4B\sigma^2}[\mathbf{e}_{(k)_N}\mathbf{e}_{(k+1)_N}]$ where $[\mathbf{e}_i]_j = \delta_{ij}$. Moreover, for every N and K such that $2K/N$ is an integer, we have that

$$\begin{aligned} \bar{P}_{o,\mathrm{cdma},N}(R\mathbf{1}) &< P_{o,\mathrm{fdma}}(R\mathbf{1}) \\ &= 1 - \exp[-\frac{1}{\lambda}(e^{KR/B} - 1)]. \end{aligned} \quad (15)$$

□

In Figure 4, we show the outage probabilities of the GMAC and FDMA frequency non-selective fading channels and the upper bound to the outage probability of the CDMA channel with $N = K = 3$. We observe that CDMA achieves a much higher rate than FDMA and is very close to the optimal.

References

[1] R. S. Cheng and S. Verdú, "Gaussian multiaccess channels with ISI: capacity region and multiuser water-filling," *IEEEIT*, vol. 39, no. 3, pp. 773–785, May 1993.

[2] M. Rupf and J. L. Massey, "Optimum sequence multisets for symbol-synchronous code-division multiple-access channels," to appear in *IEEE Transactions in Information Theory*, 1993.

[3] L. H. Ozarow, S. Shamai, and A. D. Wyner, "Information theoretic considerations for cellular mobile radio," Preprint, Mar. 1992.

[4] R. S. Cheng, "On capacities of frequency non-selective slowly-varying fading channels," in *Proc. 1993 IEEE Int. Symp. Inform. Theory*, San Antonio, TX, p. 260, Jan. 1993.

[5] S. Verdú, "The capacity region of the symbol-asynchronous Gaussian multiple-access channel," *IEEE Transactions on Information Theory*, vol. IT-35, no. 4, pp. 733–751, July 1989.

[6] R. M. Gray, *Entropy and Information Theory.* New York: Springer-Verlag, 1990.

[7] I. Csiszár and J. Körner, *Information Theory: Coding Theorems for Discrete Memoryless Systems.* London: Academic Press, Inc., 1981.

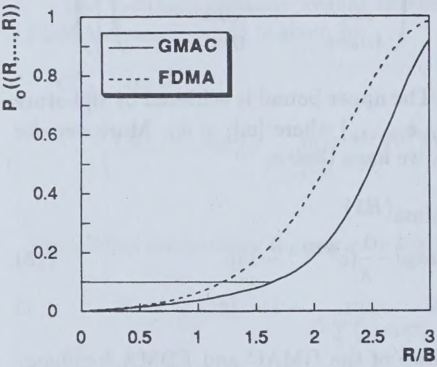


Figure 1: Outage Probability for (R,...,R) (K=2, λ=20dB)

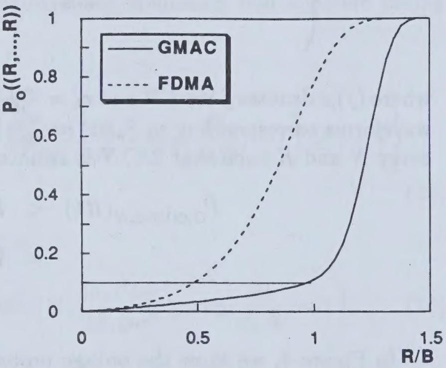


Figure 2: Outage Probability for (R,...,R) (K=5, λ=20dB)

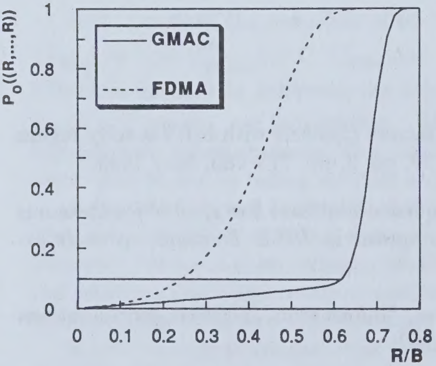


Figure 3: Outage Probability for (R,...,R) (K=10, λ=20dB)

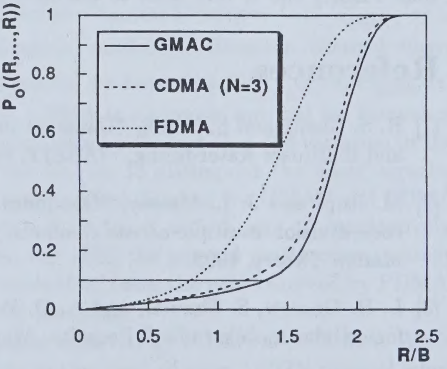


Figure 4: Outage Probability for (R,...,R) (K=3, λ=20dB)

SCHEDULED ACCESS WITH A SPACE COMPONENT

A. Ephremides
University of Maryland

ABSTRACT

In this paper we describe the spatial component of the Multiple Access problem. Among its various instances we select the one that applies to satellite channels with switchable spot beams. We describe an approach to the call admission problem in such an environment and report some results from our analysis.

SUMMARY

The classical problem of multiple access involves a model confined to the limited case of a population of users attempting to access a single receiver over a common channel. When spatial considerations are introduced the problem acquires totally new aspects. For example, a first step toward a spatially modified multiple access problem definition is to consider two receivers and a population of users trying to access either one of them. The range of communication is finite and the spatial distribution of the user population creates three subsets of users, one able to access only the first receiver, one that can access only the other receiver, and one that is in range of both. The entire gamut of multiple access questions can be asked with respect to this model, namely what are the throughput limits, how random access protocols perform (from ALOHA to collision resolution algorithms, to CSMA, etc.), and what is the information theoretic capacity of the shared channel.

We can go one step further by considering the case where there are two or more receivers and because of either mobility of the users (and/or receivers) or other physical characteristics of propagation, the connectivities vary so that at each time there are "footprints" associated with each receiver such that only users who happen to be in one of those footprints may access the receiver at that time. This situation arises in cellular applications, in shielded (in-door) wireless environments, in meteor-burst communications, and in satellite channels with switchable spot beams.

It is the last mentioned case that we focus on in this paper. We restrict attention to connection-oriented service in which users attempt to establish virtual circuits through the satellite channel. We assume a reservations-based approach in which all call-requests are assumed to be available to the system manager, who must decide whether to accept the call based on the numbers of on-going different call types and on the configuration of the system.

Typically, new circuit-switched voice or data connections are admitted to a communication system as long as resources are available. However, it was shown recently by Jordan and Varaiya [1] that the blocking of some calls when resources are available may result in a decrease of the overall blocking probability. We apply this idea of admission control to a network with a hopping-beam satellite, and we show the performance gains in terms of network revenue and overall blocking probability with respect to an uncontrolled system.

Satellites with multiple spot beams have been introduced to achieve a higher power flux density in the footprints, and to allow for frequency reuse. In order to guarantee full connectivity between all footprints, on-board switching is required. This has traditionally been done with Satellite-Switched Time-Division Multiple-Access (SS-TDMA) using a microwave switch matrix on board the satellite [2]. According to a predefined time plan, each uplink beam is periodically switched to each downlink beam. the time plan is more or less static, and, in the case of INTELSAT, is changed only a few times each year.

An increased degree of freedom can be achieved using satellites with multiple hopping beams, as e.g., the NASA Advanced Communications Technology Satellite ACTS [3]. In this case, the hopping- and switching-pattern can be changed rapidly according to traffic demand, at most once every 300 ms.

A unique feature of a hopping-beam satellite network as compared to a terrestrial network is its flexibility. The network topology can be changed almost instantaneously, which allows for an optimum on-demand resource allocation. Whereas in existing systems, such as ACTS, resources are allocated as long as they are available, we show that a more restrictive admission control policy can improve the performance of the overall network.

Note that this idea may also be applied to an integrated mobile cellular/satellite network. In this case, it has to be decided which resource type a mobile terminal should use (terrestrial or satellite), or if it should be blocked to maximize network performance.

The idea of state-dependent admission control is motivated by the work of Jordan and Varaiya [4], who recently introduced the methodology of multiple service, multiple resource (MSMR) communication networks. They showed that in some calls even if capacity is available.

In this paper, we first show that a hopping-beam satellite can be modeled as an MSMR communication network. The resources are the transponders and the switches of the satellite, and the services are point-to-point circuit-switched links between any two ground stations. We then compute the revenue, throughput and blocking probability of such a network for a given traffic model.

The main part of the paper consists of the investigation of admission control policies for such a hopping-beam satellite network. We start with a simple model of a satellite with a single hopping beam, and we then extend the analysis to satellites with multiple hopping beams. We show that it is advantageous to restrict the number of some calls that use many resources of other calls, and, if admitted, would block these calls.

We present an algorithm to compute an admission control policy which depends on the number and type of calls already in progress, and which leads to a better performance than the uncontrolled scheme. We quantify the performance gains in terms of revenue, throughput and blocking probability.

We show that the network revenue and the overall blocking probability of a hopping-beam satellite network can be improved by basing admission-control decisions on the state of the network, i.e., by rejecting some calls even in the presence of available resources. We present a method to find an optimum admission control policy for circuit-swithched links, and we quantify the improvements. Interestingly, the improvement is very limited and becomes unnoticeable as the numbers of beams, footprints, and users become larger. The analysis is computation intensive as the problem is of the combinatorial optimization type.

REFERENCES

1. S. Jordan and P. Varaiya, "Control of Multiple Service, Multiple Resource Communication Network," submitted to *IEEE Transactions on Communications*, Jan. 1991.
2. S.J. Campanella, G-P. Forcina, B.A. Pontano, and J.L. Dicks, "SS-TDMA system considerations," *COMSAT Technical Review*, Vol. 20, No. 2, pp. 335-369, Fall 1990.
3. L.C. Palmer and L.W. White, "Deman Assignment in the ACTS LBR System," *IEEE Transactions on Communications*, Vol. 38, No. 5, pp. 684-92, May 1990.
4. S. Jordan and P. Varaiya, "Throughput in Multiple SErvice, Multiple Resource Communication Networks," *IEEE Transactions on Communications*, Vol. 39, No. 8, pp. 1216-22, August 1991.
5. C.M. Barnhart, J.E. Wieselthier, A. Ephremides, "An Approach to Voice Admission Control in Multihop Wireless Networks," *Proc. of IEEE INFOCOM'93, Paper 3a.1, March 1993.*

RANDOM ACCESS WITH LARGE PROPAGATION DELAY

RAMASWAMY MURALI AND BRIAN HUGHES
Department of Electrical and Computer Engineering
The Johns Hopkins University
Baltimore, MD 21218

Abstract

Random access to a packet broadcast channel with large propagation delay is investigated. A random access protocol is presented that combines ALOHA random access with the use of forward error correction across transmitted packets. The throughput, delay, and drift behavior of this scheme is analyzed and compared to that of uncoded ALOHA. Examples illustrate that our scheme exhibits the same bistability and saturation behavior as slotted ALOHA with immediate feedback. Moreover, it is shown that forward error correction can increase throughput and reduce delay of certain system equilibrium points, but at the cost of aggravating or inducing bistability.

I. Introduction

Consider the problem of random access to a packet broadcast channel for which the feedback propagation delay is a very large multiple of the packet transmission time (or slot length). This situation arises in optical networks, where the data rates are very high (Gbs/sec) [1].

In classical random access schemes such as ALOHA [2], [3], users send packets immediately upon receipt and wait for an acknowledgement before either retransmitting the packet or waiting for the next packet. However, as pointed out in [1], these schemes were developed for channels such as the satellite channel, for which the feedback propagation delay is of the order of 12 times the slot length. This raises the question of whether traditional random access schemes are inherently unsuitable for channels where the propagation delay is several hundred slot lengths.

In the presence of large propagation delays it is desirable to use error-control coding, which can increase the probability of successful transmission and hence reduce delay. Schemes involving coding across packets with *non-bursty* sources and *no-feedback* have been examined in [4], [5]. The stability properties of slotted ALOHA with *multi-packet* reception and *immediate* feedback was investigated in [6]. The application of multiuser coding in random access schemes for the case of *immediate* feedback and *user-priority* was considered in [7]. Recently, bounds on the delay of schemes, for the case of *bursty* sources and *large* feedback delays, were developed in [1].

In [8], a random access scheme with coding across transmitted packets was presented. Based on an approximate model the throughput, delay, and drift behavior was examined using drift vector-field plots. In this paper, we extend upon our earlier work by (a) deriving a more rigorous model for our scheme, (b) developing analytical tools to determine the location and nature of the equilibrium points of our scheme, and (c) examining the dependence of the dynamic behavior of our scheme on code parameters.

Our work differs from earlier work [1], [6]-[7] in three important aspects. First, we qualitatively assess the impact of large propagation delay on the throughput, delay, and drift of slotted ALOHA. Second, we focus on the use of coding in slotted ALOHA, for the case of large propagation delay, and determine its impact on the throughput, delay, and drift performance. Third, we consider *finite* user populations.

The rest of the paper is organized as follows. In Section II, we outline a scheme that combines features of ALOHA random access with users coding across packets. In Section III, throughput, delay, and drift equations are derived for this scheme. In Section IV, we analyze the throughput, delay, and drift of our scheme and compare its performance with that of slotted ALOHA. Finally, in Section V we briefly summarize our conclusions.

II. Random Access with Coding

A. Packet Generation

Each user employs an (n,k) Reed-Solomon (RS) code over GF(2^p), where $n = 2^p$. For the purposes of coding, the L data bits in a message are regarded as a sequence of $l = L/p$ symbols from GF(2^p). When a new message arrives, its contents are encoded into a series of packets according to the following procedure.

(1) The l symbols in the message are copied into a $k \times l/k$ array, as illustrated in Fig. 1.

(2) Next the array is augmented with $n-k$ additional rows using the (n,k) RS code. The last $n-k$ symbols in each column are the parity-check symbols of the first k symbols in that column.

(3) Each *row* of the array is now packaged as a separate packet by adding a header and a cyclic redundancy checksum (CRC) for collision detection. In addition to providing for identification and synchronization, the header also specifies the row from which the packet was formed.

This procedure creates n packets which are henceforth referred to as *code packets.* The first k code packets, which contain the original message, will be called *information packets;* the remaining $n-k$ will be called *parity packets.*

B. Packet Transmission

We assume that time is subdivided into slots of duration equal to the time required to transmit one code packet. Users are restricted to transmit their code packet within slot boundaries.

We assume that each user adopts the following *uniform transmission policy.* The n code packets, that represent a message, are transmitted in sequence in n slots that are randomly selected from the next $m \geq n$ slots, with all $\binom{m}{n}$ choices being equally likely. The m contiguous slots within which users send their n code packets will be called a *frame.* In this paper, we will assume that users are *frame-synchronized.*

C. Packet Collisions and Retransmission

When a single packet is transmitted in a slot, we assume that it is always received correctly. However, when two or more packets are transmitted in the same slot, all are destroyed by the collision and declared *erasures* by the receiver. Since the (n,k) RS code has minimum distance $n-k+1$, in our scheme, the receiver can recover the entire message of a user when any k or more code packets are received correctly. When fewer than k packets are received successfully, the receiver requests a retransmission.

Following transmission of the n code packets, each user waits for acknowledgement from the receiver, informing of whether or not the message was successfully received.

We assume that users receive this acknowledgement N slots after the commencement of transmission of the message.

We consider the simple retransmission scheme, called *frame retransmission* [8]. Upon receiving the retransmission request, the user selects a frame at random from one of the next $K-1$ frames, $K>1$, where each choice is equally likely. The user then sends all n code packets again in this frame according to the uniform transmission policy.

III. Throughput, Delay, and Drift

We now develop an approximate Markovian model for our random access scheme. Using this model, we derive expressions for throughput, delay, and drift. These results are then specialized to the classic slotted ALOHA algorithm.

A. An Approximate Model

In our scheme, users wait N slots before learning whether they must retransmit. An exact analysis would therefore require a state description that includes the channel history for the preceding N slots. In this paper, we make no attempt to conduct such an analysis; rather, our approach is to present an approximate model which captures some of the main features of the scheme and yet is simple enough to permit a complete analysis. Our model is based on four approximations. The first three are as follows.

(A1) The feedback propagation delay (in frames) is modeled as a geometric random variable with mean $1/p_w = N/m$. The sum of the feedback propagation delay and the retransmission delay is similarly modeled as geometric with mean $1/p_r = N/m + K/2$.

(A2) For a given user, the sequence of code packet erasures within a frame is a Bernoulli sequence.

(A3) When several users transmit in the same frame, the events that these users are successful are independent and identically distributed.

Approximation (A1) has been used in the analysis of ALOHA for satellite channels [9]. (A2) and (A3) are reasonable approximations when $m >> n$ and the total number of users attempting transmission in a frame is not small.

The assumptions above suggest the following simple Markovian model. Each user is in one of three states: *fresh* (F), *backlogged* (B), or *waiting* (W). In each frame, users in the fresh state attempt transmission of a new message with probability p_o. With probability P_s, this transmission is successful and the user enters the waiting state; with probability P_e, the transmission is unsuccessful and the user enters the backlogged state. The probability $P_s = 1 - P_e$ depends upon the state of all users and will be defined later. While a user is in states B and W, new message arrivals are *blocked.* With probability p_w in each frame (from (A1)), users in state W learn that their previous transmission was successful, whereupon they return to F. With probability p_r in each frame, users in B retransmit their message and again split into states B and W.

The three states, and the possible transitions among them, are summarized in Fig. 2. Defining N_α^l to be the number of users in state $\alpha \in \{F, B, W\}$ during frame l, we see that $\underline{X}^l = (N_F^l, N_B^l)$, $l = 1, 2, 3, \ldots$, constitutes a discrete-time Markov chain.

Note that, when conditioned on $\underline{X}^l = (n_F, n_B)$, the total number of users that attempt transmission in frame l can be written as $T^l = T_F^l + T_B^l$, where T_F^l and T_B^l are independent, Binomial random variables with parameters (n_F, p_o) and (n_B, p_r), respectively. Our final approximation is given below.

(A4) Given $\underline{X}^l = (n_F, n_B)$, the quantity T^l can be modeled as a Poisson random variable with mean $n_F p_o + n_B p_r$.

We now reparametrize the problem, as in [10]. In a given frame, let r_α denote the fraction of the M users that are in state $\alpha \in \{F, B\}$, so that $0 \leq r_F, r_B \leq 1$ and $r_F + r_B \leq 1$. Hereafter, we regard $\underline{r} = (r_F, r_B)$ as the system state.

Observe that, when all users are in state F, the arrival rate of fresh information packets per slot is given by $\Lambda_F = M p_o k/m$. Similarly, when all users are in B, the rate at which backlogged information packets are retransmitted is $\Lambda_B = M k p_r/m$ packets per slot. For any given state $\underline{r}$, the total arrival rate of information packets (fresh and backlogged) is therefore

$$\Lambda(\underline{r}) = r_F \Lambda_F + r_B \Lambda_B. \tag{1}$$

It then follows that the arrival rate of code packets per slot is $G(\underline{r}) = \Lambda(\underline{r})\ (n/k)$.

For a given $\underline{r}$, we next derive an expression for the probability, $P_s(\underline{r})$, that a transmitted message (fresh or backlogged) is successfully received. When $\underline{X}^l = (r_F M, r_B M)$, then (A4) implies that T^l is Poisson distributed with mean $\mu = \Lambda(\underline{r})\ (m/k)$.

In our scheme, the probability that a single code packet of a user is erased (hit) by the transmitted code packets of any other specific user is easily seen to be $P_h = n/m$. If J other users attempt transmission, the probability that a given user's code packet is erased is

$$P_e(J) = 1 - (1 - P_h)^J. \tag{2}$$

Since a message is successfully received if and only if at most $n - k$ of the n code packets are erased, the probability that a given user's message is successfully received when J other users attempt transmission is

$$P_{succ}(J) = \sum_{i=0}^{n-k} \binom{n}{i} P_e(J)^i \ (1 - P_e(J))^{n-i}. \tag{3}$$

In obtaining (3), we have used (A2). It follows that if the system state is $\underline{r}$, we have

$$\begin{aligned} P_s(\underline{r}) &= \sum_{J=0}^{\infty} \frac{e^{-\mu}\ (\mu)^J}{J!}\ P_{succ}(J), && (4) \\ &= \sum_{J=0}^{\infty} \frac{e^{-\mu}\ (\mu)^J}{J!} \sum_{i=0}^{n-k} \binom{n}{i} \left[1 - (1 - P_h)^J\right]^i \ \left[(1 - P_h)^J\right]^{n-i}. && (5) \end{aligned}$$

We are primarily interested in the case where M, N, and m are large but M/m and N/m are moderate. In this case, as demonstrated in [11, pg. 656], (5) can be approximated by

$$P_s(\underline{r}) = \sum_{i=0}^{n-k} \binom{n}{i} \left[1 - \exp(-G(\underline{r}))\right]^i \ \left[\exp(-G(\underline{r}))\right]^{n-i}. \tag{6}$$

In the following, by $P_s(\underline{r})$ we refer to the expression in (6). Also, when $\underline{r}$ is given, (A3) implies that transmitted messages from different users succeed or fail independently, with common success probability $P_s(\underline{r})$.

B. Throughput

For a given state $\underline{r}$, the *throughput*, $S_{out}(\underline{r})$, is the expected number of successful information packets per slot. It is the product of the rate at which information packets arrive, and the probability that these packets are successful. For our scheme, the probability that a transmitted information packet is received successfully is the same as the probability that the message (of which it is a part) is received successfully. Hence, using (1) and (6), we have

$$S_{out}(\underline{r}) = \Lambda(\underline{r})\, P_s(\underline{r}). \tag{7}$$

C. Delay

We define the *delay* of a message to be the number of slots that elapse between the start of the frame in which the message is first transmitted, and the start of the frame in which the message is transmitted successfully. Note that this definition ignores both the forward propagation delay ($N/2$ slots) and the transmission delay (not more than m slots) of the last frame.

If the system does not drift significantly from state $\underline{r}$ during message retransmissions, the expected delay of a packet is then given by

$$D(\underline{r}) = (N + mK/2)\left(\frac{1 - P_s(\underline{r})}{P_s(\underline{r})}\right) \text{ slots.} \tag{8}$$

D. Drift

We want to examine how the system state tends to evolve with time. To this end, we generalize the notion of *drift* [10] to our three-state model.

For $\alpha \in \{F, B\}$, the *drift* into state α, $d_\alpha(\underline{r})$, is the expected change in r_α during a slot. Thus, the drift into a state is proportional to the difference between the rate at which users enter the state and the rate at which users leave the state. We refer to $\underline{d}(\underline{r}) = (d_F(\underline{r}), d_B(\underline{r}))$ as the *drift field.*

The rate at which users enter the backlogged state is $r_F\ \Lambda_F\ P_e(\underline{r})/k$, since $r_F\ \Lambda_F$ is the rate of generation of fresh information packets and $P_e(\underline{r})$ is the probability that a transmitted message is unsuccessful. The rate at which users leave the backlogged state is $r_B\ \Lambda_B\ P_s(\underline{r})\ /k$. Thus, we have

$$\begin{aligned} d_B(\underline{r}) &= (kM)^{-1}[r_F\ \Lambda_F\ P_e(\underline{r}) - r_B\ \Lambda_B\ P_s(\underline{r})] \\ &= (kM)^{-1}[r_F\ \Lambda_F - \Lambda(\underline{r})\ P_s(\underline{r})]. \end{aligned} \tag{9}$$

Similarly, we can show that

$$d_F(\underline{r}) = (kM)^{-1}[(1 - r_F - r_B)\ \Lambda_W - r_F\ \Lambda_F], \tag{10}$$

where $\Lambda_W = Mkp_w/m$.

The drift $\underline{d}(\underline{r})$ constitutes a *vector field* on the state triangle $r_F, r_B \geq 0$, $r_F + r_B \leq 1$. At each point $\underline{r}$ in this triangle, $\underline{d}(\underline{r})$ describes the expected movement of the system per slot. An *equilibrium point,* $\underline{r}_e$, for this system is one that satisfies

$$d_F(\underline{r}_e) = d_B(\underline{r}_e) = 0\ . \tag{11}$$

An equilibrium point $\underline{r}_e$ is *locally stable* if, on average, small pertubations move the system back to the equilibrium point. Mathematically, this is equivalent to the eigen values λ_1, λ_2 of the Jacobian matrix

$$C = \begin{bmatrix} \frac{\partial d_F(\underline{r}_e)}{\partial r_F} & \frac{\partial d_F(\underline{r}_e)}{\partial r_B} \\ \frac{\partial d_B(\underline{r}_e)}{\partial r_F} & \frac{\partial d_B(\underline{r}_e)}{\partial r_B} \end{bmatrix}$$

satisfying the condition $\mathrm{Re}(\lambda_i) < 0$, $i = 1, 2$. If an equilibrium point is not locally stable, it could either be a *saddle* point ($\mathrm{Re}(\lambda_1)\mathrm{Re}(\lambda_2) < 0$) or else *locally unstable* ($\mathrm{Re}(\lambda_i) > 0$, $i = 1, 2$).

It is instructive to consider the two conditions in (11) separately. Let the *fresh equilibrium contour* be the locus of all points $\underline{r}$ in the triangle that satisfy $d_F(\underline{r}) = 0$. Similarly, let the *backlog equilibrium contour* consist of all points in the triangle that satisfy $d_B(\underline{r}) = 0$. Equilibrium points then lie at the intersections of these two contours.

From (10), the fresh equilibrium contour is the line

$$r_F(\Lambda_W + \Lambda_F) + r_B\Lambda_W = \Lambda_W \,. \tag{12}$$

Next, note that $P_s(\underline{r})$ in (6) depends on $\underline{r}$ only through $\Lambda = \Lambda(\underline{r})$; we therefore abuse notation slightly by writing $P_s(\Lambda) = 1 - P_e(\Lambda)$ to display this dependence. With this definition, the backlog equilibrium contour can be recognized as that part of the parametric curve

$$\underline{r}_T(\Lambda) = (\Lambda P_s(\Lambda)/\Lambda_F, \Lambda P_e(\Lambda)/\Lambda_B)\ ,\ \Lambda \geq 0\ , \tag{13}$$

which lies in the state triangle. To see this, suppose for a given $\underline{r}$ that $\Lambda = \Lambda(\underline{r})$. Then $d_B(\underline{r}) = 0$ implies $r_F\Lambda_F = \Lambda P_s(\Lambda)$ and hence $r_F = \Lambda P_s(\Lambda)/\Lambda_F$. It then follows that $r_B = (\Lambda - r_F\Lambda_F)/\Lambda_B = \Lambda P_e(\Lambda)/\Lambda_B$. The significance of (13) lies in the observation that, for all choices of Λ_B and Λ_F, the backlog equilibrium contour is a rescaled version of the curve $\underline{r}_T(\Lambda) = (\Lambda P_e(\Lambda), \Lambda P_s(\Lambda))$. Thus, the shape of the curve is determined solely by the code parameters n and k, whereas the scale of the curve is determined by the arrival rates Λ_B and Λ_F.

E. Slotted ALOHA

When $n = k = m = 1$, there is no error-control coding and a frame coincides with a slot; hence, our random access scheme reduces to the classic slotted ALOHA algorithm. The preceding results then specialize to slotted ALOHA with long propagation delays. The throughput (7) then reduces to the familiar formula

$$S_{out}(\underline{r}) = (r_F\Lambda_F + r_B\Lambda_B)\,\exp\left[-(r_F\Lambda_F + r_B\Lambda_B)\right]. \tag{14}$$

and the average packet delay (8) becomes

$$D(\underline{r}) = (N + K/2)\left(\frac{1 - P_s(\underline{r})}{P_s(\underline{r})}\right). \tag{15}$$

The drift field is as given in (9) and (10).

IV. Numerical Results

In this section, we investigate the throughput, delay, and drift of our scheme, and compare its performance with that of slotted ALOHA.

A. Fundamental Parameters

In order to make a fair comparison, we wish to evaluate the performance of these schemes at the same arrival rate in *bits per second*, and the same propagation and retransmission delays in *seconds*. To this end, observe that a message will be transmitted as one packet in slotted ALOHA, whereas it will be subdivided into k information packets in our scheme, where k is the dimension of the code. Thus, a packet and a slot are k times larger in ALOHA than in our scheme. It follows that a slotted ALOHA system with parameters N and K will be roughly comparable to the coded scheme with parameters $N' = kN$ slots and $K' = kK/m$ frames. With this choice of N' and K', the other system parameters — M, Λ_F, Λ_W, and $\Lambda_B = \Lambda_W/(1 + K/2N)$ — will be the same in both schemes.

In order to determine the throughput, delay, and drift of our scheme, the constants Λ_F, Λ_W, and Λ_B (or alternately K/N) must be specified. We refer to $\Omega = \{\Lambda_F, \Lambda_W, K/N, n, k\}$ as the *parameter set* of our scheme. Likewise, the parameter set for slotted ALOHA can be defined, with $n = k = 1$. In all the comparisons that follow, the system parameters — Λ_F, Λ_W, and K/N — are for slotted ALOHA.

We remark here that our scheme will generally require more overhead, in the form of header information and CRCs, than ALOHA. The preceding comparison is therefore valid only when the overhead of both schemes is negligible in comparison to packet length. However, this comparison is easily adapted to the case when the overhead is nonnegligible.

B. Throughput-Delay-Drift

We determine the *equilibrium* throughput, delay, and drift performance of our scheme (or slotted ALOHA), with parameter set Ω, by plotting the fresh (F_Ω) and backlog (B_Ω) equilibrium contours using (12) and (13), respectively. At an equilibrium point $\underline{r}_e$, the throughput is $r_F^e \Lambda_F$, while the delay (excluding the forward propagation delay) is the ratio of the number of backlogged users ($r_B^e M$) to throughput. The nature of the equilibrium point (i.e, stable, unstable, or saddle) can be determined using the procedure in section III.D. We then compare the performance of our scheme with that of slotted ALOHA by contrasting the throughput, delay, and nature of their equilibrium points.

In order to illustrate the performance of our scheme, we consider three choices of parameter sets: (1) $\Omega_1 = \{0.4, 0.4, 0.01, 8, 2\}$, (2) $\Omega_2 = \{0.2, 1.2, 0.01, 8, 2\}$, and (3) $\Omega_3 = \{0.3, 1.8, 0.01, 8, 2\}$. In Figs. 3, 4, and 5, the fresh and backlog equilibrium contours of our scheme are plotted for cases $1, 2, 3$, respectively. Also, shown in Figs. 3, 4, and 5 are corresponding contours for slotted ALOHA. Let us denote by Ω_1', Ω_2', and Ω_3' the parameter sets for slotted ALOHA. Observe that, in Figs. 3, 4, and 5, the fresh equilibrium contour is a common straight line for the two schemes.

From Fig. 3 and Table 1, it follows that both our scheme (Ω_1) and slotted ALOHA (Ω_1') have a single locally stable equilibrium point. Also, we note that, compared to slotted ALOHA, our scheme attains equilibrium with higher r_F^e and lower r_B^e, which implies higher throughput and lower backlog (hence delay). Thus, in this case, our scheme provides higher throughput and lower delay at its equilibrium point when compared to slotted ALOHA, without altering the nature of the equilibrium point. Notable is the fact that in both schemes a significant fraction of users (roughly half) are in the waiting state.

From Fig. 4 and Table 1, we note that, while slotted ALOHA (Ω_2') has a single locally stable equilibrium point, our scheme (Ω_2) has three equilibrium points. Two of them are locally stable and one a saddle point. Of the two locally stable equilibrium points one of them is desirable with high r_F^e and low r_B^e, while the other is not since it leads to very low throughput and large backlog. Thus, in this case, although our scheme has a desirable stable equilibrium point (with comparable throughput and delay to that achieved by slotted ALOHA), it exhibits bistability while slotted ALOHA does not.

Finally, from Fig. 5 and Table 1, we note that both schemes have a single stable equilibrium point. However, while our scheme (Ω_3) reaches saturation with low throughput and high backlog, slotted ALOHA (Ω_3') achieves high throughput and low backlog at its equilibrium point. Thus, for this choice of sets, our scheme performs significantly poorer than slotted ALOHA.

Table 1: Nature of equilibrium points for various parameter sets

Ω	(r_F^e, r_B^e)	λ_1	λ_2	Nature
$\{0.4, 0.4, 0.01, 8, 2\}$	$(0.481, 0.038)$	-0.39	-0.66	Stable
$\{0.4, 0.4, 0.01, 1, 1\}$	$(0.444, 0.111)$	-0.39	-0.65	Stable
$\{0.2, 1.2, 0.01, 8, 2\}$	$(0.852, 0.005)$	-1.14	-1.21	Stable
	$(0.508, 0.407)$	$+0.38$	-1.19	Saddle
	$(0.012, 0.985)$	-0.18	-1.19	Stable
$\{0.2, 1.2, 0.01, 1, 1\}$	$(0.830, 0.031)$	-0.97	-1.20	Stable
$\{0.3, 1.8, 0.01, 8, 2\}$	$(0.000, 0.999)$	-0.29	-1.80	Stable
$\{0.3, 1.8, 0.01, 1, 1\}$	$(0.809, 0.055)$	-1.14	-1.80	Stable

We next investigate the performance of our scheme for other choices of parameter sets. We do so by considering various cases, depending on which of the set-components are varied.

1. Λ_F and Λ_W: From (12) and (13), we note that variation of Λ_F and Λ_W alters the abscissa of F_Ω and changes the scale of B_Ω, respectively. Thus, the location (and possibly number, nature) of the equilibrium points of the scheme are altered.

2. K/N: Since N is fixed by the channel, we need consider only changes in K. Variation in K affects Λ_B, which controls the vertical scale of B_Ω. Note that, F_Ω is not affected. Thus, changing K has the effect of altering the location and number (possibly) of equilibrium points along F_Ω.

3. n and k: Variation of code parameters alters only B_Ω, as illustrated in Fig. 6. Observe that B_Ω contracts rapidly along the vertical axis as the blocklength is increased. The contraction of B_Ω with increasing n has several important consequences. First, larger blocklength codes achieve higher throughput and lower delay at the desirable stable equilibrium point. Second, large n can induce bistability or lead to saturation. Third, as n is increased the separation between the desirable stable equilibrium and the saddle point is reduced. This suggests that smaller statistical fluctuations may cause the scheme (with larger n) to transit from the desirable stable equilibrium to saturation.

V. Summary

We conclude that our scheme exhibits the same qualitative behavior as slotted ALOHA with immediate feedback. Depending on the choice of parameter sets, the use of coding in ALOHA random access either increases throughput and reduces delay or else induces bistability, even leading to saturation. As blocklength increases, lower delay at the desirable equilibrium point is achieved at the cost of transiting to saturation at lower backlogs. Thus, for more favorable performance, improved retransmission policies with short blocklength codes are desirable.

Acknowledgement

This research was supported by the U. S. Army Research Laboratory and the Army Research Office under grant no. DAAL03-89-K-0130, and by the National Science Foundation under grant no. NCR-9217457.

References

[1] B. Hajek, B. S. Tsybakov, and N. B. Likhanov, "On the delay in a multiple access system with large propagation delay," in *Abstracts of Papers, IEEE Int. Symp. Information Theory*, San Antonio, Texas, Jan. 17-22, 1993.

[2] N. Abramson, "The ALOHA system: Another alternative for computer communications," *Proc. 1970 Fall Joint Comput. Conf., AFIPS Conf.*, vol. 37, pp. 281-285, 1970.

[3] L. G. Roberts, "ALOHA packet system with and without slots and capture," *Computer Communications Review*, vol. 5, pp. 28-42, April 1975.

[4] J. L. Massey and P. Mathys, "The collision channel without feedback," *IEEE Trans. Inform. Theory*, vol. IT-31, pp. 192-204, March 1985.

[5] B. S.Tsybakov and N. B. Likhanov, "Packet switching in a channel without feedback," *Prob. Peredachi Inform.*, vol. 19, no. 2, pp. 69-84, Apr.-June, 1983.

[6] S. Ghez, S. Verdu, and S. C. Schwartz, "Stability properties of Slotted ALOHA with multipacket reception capability," *IEEE Trans. on Automatic Control*, vol. 33, no. 7, pp. 640-649, July 1988.

[7] G. Thomas, "Application of multiuser coding to random access protocols," *IEEE Trans. on Communications*, vol. COM-36, no. 8, pp. 983-986, August 1988.

[8] R. Murali and B. L. Hughes, "The effect of coding on the stability of random access systems," *Proc. 1993 Conf. on Inform. Sci.and Sys.*, Johns Hopkins University, Baltimore, M.D., March 1993.

[9] L. Kleinrock and S. S. Lam, "Packet switching in a multiaccess broadcast channel: Performance evaluation," *IEEE Trans. on Communications*, vol. COM-23, no. 4, pp. 410-423, April 1975.

[10] A. B. Carleial and M. E. Hellman, "Bistable Behavior of Aloha-Type Systems," *IEEE Trans. on Communications*, vol. COM-23, no. 4, pp. 401-410, April 1975.

[11] M. B. Pursley, "Frequency-hop transmission for satellite packet switching and terrestrial packet radio networks," *IEEE Trans. Inform. Theory*, vol. IT-32, no. 5, pp. 652-667, Sept. 1986.

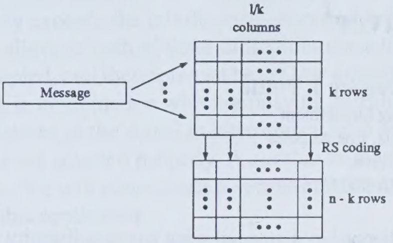

l/k
columns
Message
k rows
RS coding
n - k rows

Figure 1. Code packet generation.

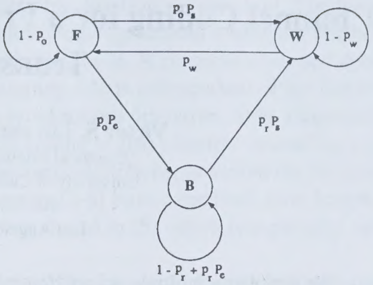

$1 - p_o$
F
$p_o p_s$
W
$1 - p_w$
p_w
$p_o p_e$
$p_r p_s$
B
$1 - p_r + p_r p_e$

Figure 2. State transition diagram.

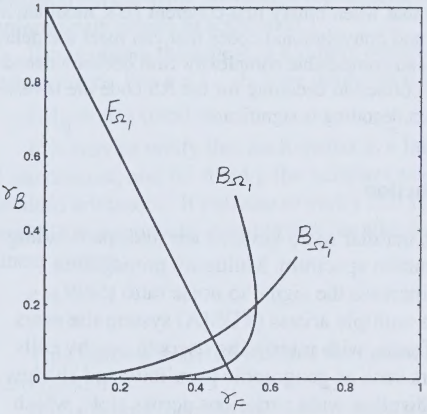


Figure 3. Equilibrium contours for $\Lambda_F = \Lambda_W = 0.4,\ K/N = 0.01$.

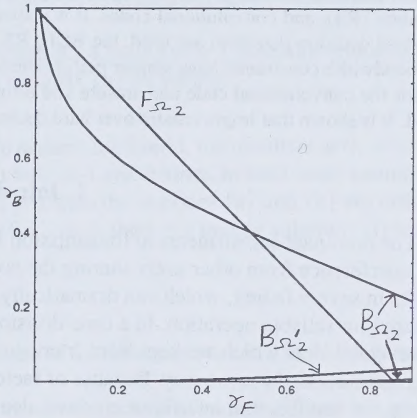


Figure 4. Equilibrium contours for $\Lambda_F = 0.2,\ \Lambda_W = 1.2,\ K/N = 0.01$.

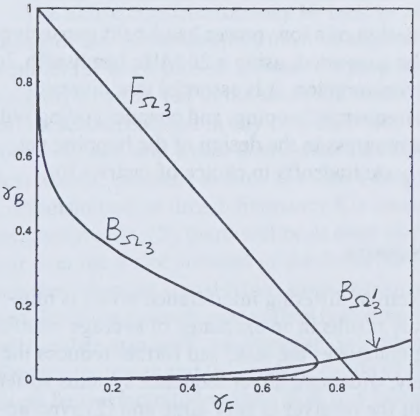


Figure 5. Equilibrium contours for $\Lambda_F = 0.3,\ \Lambda_W = 1.8,\ K/N = 0.01$.

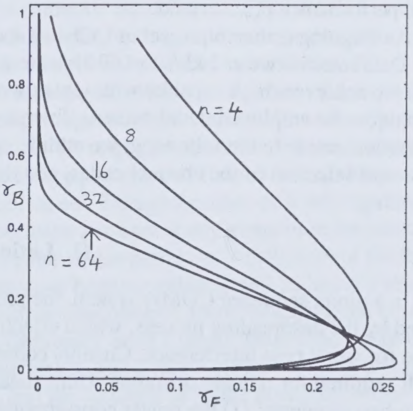


Figure 6. Backlog equilibrium contours for $\Lambda_F = \Lambda_B = 1,\ k/n = 1/4$.

Channel Coding for a Frequency-Hopped Wireless Transceiver[1]

Victor S. Lin and Gregory J. Pottie
Electrical Engineering Department
University of California, Los Angeles
405 Hilgard Ave.
Los Angeles, CA 90024

Abstract - We consider a multiple-access frequency-hopped system employing antenna diversity and error correction coding. A construction based on orthogonal Latin squares is used for the hopping patterns to improve code performance in the presence of interference. The measures of interest are the system performance, delay and decoder complexity, providing a basis for comparing BCH, Reed-Solomon (RS), and convolutional codes. It is shown that when binary non-coherent FSK modulation and hard decision detection are used, the BCH, RS, and convolutional codes that can meet the delay and bandwidth constraints have similar performance and comparable complexity. Soft decision decoding for the convolutional code and erasure and error correction decoding for the RS code are investigated. It is shown that improvement over hard decision decoding is significant.

I. Introduction

The dominant impairments to transmission in cellular radio systems are multipath fading and interference from other users sharing the common spectrum. Multipath propagation results in severe fading, which can dramatically increase the signal to noise ratio (SNR) required for reliable operation. In a time-division multiple access (TDMA) system the users are assigned slots which are kept from frame to frame, with interfering users in nearby cells assigned slots in the same way. Because of factors such as geographic proximity and shadow fading, the result is that interference power also displays wide variations across slots, which further degrades the 90% to 99% worst case performance. The conventional solution of restricting the frequency re-use between cells significantly reduces capacity. Since the re-use distance must be made large enough to allow sufficient margin against multipath fading, a high performance price is paid.

An ongoing research project at UCLA is the design of a low-power hand-held transceiver [1]. Data rates between 2 kb/s to 160 kb/s are to be supported, using a 26 MHz bandwidth. In order to achieve a high capacity with low power consumption, it is essential that diversity techniques be employed. Dual antenna diversity, frequency hopping, and channel coding will be implemented. In the following, we outline our progress in the design of the hopping patterns and selection of the channel codes, and the basic tradeoffs in choice of metrics for decoding.

II. Latin Squares

In a direct sequence CDMA system, the problem of differing interference levels is mitigated by the despreading process, which effectively results in an exchange of average interference for worst case interference. Channel coding completes the task, and further reduces the SNR required for reliable operation. Unfortunately, wideband direct sequence systems suffer from two problems: (1) the power consumption of the receiver is very large and (2) synchro-

---

[1] This work supported in part by AT&T Bell Laboratories Mathematical Sciences Research Center and ARPA contract DAAB07-92-R-C977.

nous access is not possible on the up-link, since the chip duration is very short. The latter problem results in decreased capacity, since the interference generated within the cell considerably exceeds the interference generated in neighboring cells. A frequency hopped system can alleviate both of these difficulties since the hopping rate is independent of the bandwidth occupied, and therefore can be set low enough to synchronous operation. Slow frequency hopping in combination with interleaving and channel coding is also effective in dealing with variations in the signal to interference ratio among successive hops, provided the hopping patterns are selected properly. A number of recent papers [2-4] have examined slow-hopped systems. We will concentrate a simple construction introduced in [5] which is especially suitable for this application.

A latin square of order n is an n x n matrix with entries from a set R of n distinct elements, say $R=\{0,1,...,n-1\}$ such that each row and column contains every element of R exactly once. Two Latin squares are said to be orthogonal if the n^2 ordered pairs (i,j), where i and j are the entries from the same position in the respective squares, exhaust the n^2 possibilities, i.e., every ordered pair occurs exactly once.

If n is prime, then there is a simple construction for a family of n-1 mutually orthogonal latin squares. For a = 1,...,n-1 we define an n x n matrix {a} by setting

$\{a\}_{ij} \equiv ai+j \pmod{n}$

It is easy to verify that each matrix is a latin square: for fixed i, the numbers ai+j, j=0,...,n-1 are distinct, and for fixed j, the numbers ai+j, i=0,..,n-1 are distinct, in both cases assuming modulo arithmetic. It remains to verify that if $a \neq b$, then the matrices {a} and {b} are orthogonal. This amounts to showing that for all c,d = 0,1,...,n-1, there is a unique solution (i,j) to the linear system

$c \equiv ai + j$

$d \equiv bi + j$

This follows from the fact that the determinant of this linear system is a-b, which is non-zero (mod n).

If n is prime, then the residues modulo n form the finite field of order n. The above construction extends to finite fields of prime power order in the natural way: now R is the set of field elements, and for every non-zero field element a, we define a matrix {a} with the ijth entry $\{a\}_{ij} = ai+j$, where addition and multiplication take place in the field.

The above construction may be used in a synchronous cellular radio system. That is, the timing of the slot boundaries must be aligned across cells, with guard bands potentially required for large cells to account for propagation delays from nearby cells. Each cell is assigned a matrix {a} in the set of n-1 mutually orthogonal latin squares. Then up to n users can be accommodated in any cell, each user being assigned a field element. For example, suppose n=8, a=5, and a user is assigned the element 7. We might interpret the rows as frequency slots and the columns as time slots for a frequency hopped system. Then at time 0, frequency 3 is transmitted, at time 1 frequency 6 is transmitted, etc. Except for other cells which also are assigned matrix {5} there will be at most one collision per user at any given time/frequency pair over the 8 slot duration of the frame. If a regular re-use pattern of 7 cells is used for the matrices, then the signals from users sharing the same hopping patterns will be heavily attenuated. In this way, each user suffers collisions from an ensemble of users occupying the nearby cells, rather than with one dominant interferer. Obviously, with a larger value of n, the re-use distance increases and the randomization of interference hits is more complete. Also, the rows might be reinterpreted as time slots in a frame, and the columns as frame positions within a superframe for a pure TDMA system.

The construction allows each mobile to quickly generate its own time/frequency hopping pattern given minimal information transfer from the base station. The only information required, given that n and the primitive polynomial p(x) are known, is the matrix number, a, and the field element, e. Then if i is the row number, the column corresponding to i is just $j \equiv e + ai \pmod{p(x)}$. Hence, the overhead involved in communicating the hopping pattern is negligible. Note that we cannot get every possible value of n using this construction. However, primes and prime powers are dense for moderate n, affording a wide range of choices.

A very similar construction has recently been proposed by Wyner [6] using a combinatorial configuration constructed by Roche [7] for use in wideband multi-tone systems, to assure that users in nearby cells interfere in at most one tone.

The effectiveness of the Latin squares construction was illustrated in [5] for a slow-hopped DQPSK system. Simulations were conducted in combination with rate-1/2 and rate-1/4 convolutional codes with 32 states, for a multi-cell system using the same propagation assumptions as in [8]. It was further assumed that Rayleigh fading was independent between frequency hops, and that channel parameters were estimated from the data. The results indicate that the system capacity without some randomization of interferers is quite poor, but with the Latin squares approach capacity increased beyond that reported for a DS-CDMA system in [8], with considerably lower decoding delay. The addition of dual antenna diversity further improved performance.

Simulations were also conducted to examine performance in flat fading conditions. The results in this case are particularly interesting in illustrating what we refer to as interferer diversity. Even though the desired signal level is now at the same level in each slot, the interference levels are independent, since a different set of interferers in encountered in each slot. Thus, even in flat-fading conditions, the interference is independently fading, leading to independent signal to interference ratios. Therefore channel coding continues to provide a significant improvement in performance, effectively averaging over the SNR variations while providing coding gain. This is of considerable interest in handheld transceiver applications, since fading can be flat over tens of MHz in some indoor situations. In these cases, frequency hopping in itself would not be sufficient to achieve diversity benefit; some randomization of interference is also required. Consequently, the Latin squares construction will be incorporated in the final design.

III. Channel Code Selection

Fig. 1 outlines the system model used in the investigation of alternative channel coding schemes. The input bit stream is encoded into the coded q-bit symbol sequence of a block code or a convolutional code and the resulting coded sequence with a rate of f_c symbols/s is fed to the interleaver. The interleaver rearranges the coded sequence in such a way that the fading channel with burst error characteristic is transformed into a channel having independent errors. The coded symbol at the output of the interleaver, to be transmitted with a rate of $f_c \times q$, is assigned to one of the two binary orthogonal FSK signals.

The frequency hopping binary FSK signal is output from a frequency hopper where the hopping local signal is multiplied to the binary FSK signal. A slow frequency hop scheme is assumed in which a hop is made in each slot interval consisting of a few symbols. It is assumed that the received signal experiences independent Rayleigh fading slot-by-slot, and that it is perturbed by additive white Gaussian noise. While this model does not yield a capacity estimate, it is sufficient to evaluate the relative effectiveness of coding schemes in a situation where we in fact expect the SNR to be independent from hop to hop as a result of the use of the Latin squares construction.

The receiver dehops the signal to obtain the received binary FSK signal. Non-coherent detection is used for the reception, because the handheld system is designed to allow fast hopping. The receiver employs dual antenna diversity with postdetection equal gain combining to achieve diversity. This was found to outperform selection diversity, and in any case when the channel SNR cannot be estimated with any accuracy for a slot (e.g., fast hopping), two complete receiver branches are required to make antenna diversity effective. Thus, equal gain combining comes essentially for free. We assume that the signals received at the two antennas have low correlation.

As a result of interleaving/deinterleaving and frequency hopping, errors within a code word appear to be independent. We assume that a convolutional interleaver is used because it adds the minimal delay. In this section, we consider hard decision decoding for convolutional codes and error correction for block codes. Later, we examine soft decision decoding for convolutional codes and erasure and error correction for block codes.

In order to compare the various coding options in terms of BER performance, we assume ideal interleaving so that the channel is memoryless. For non-coherent BFSK modulation with ideal interleaving, the error rate performance over a frequency-nonselective, slowly fading channel is

$$P_2 = \frac{1}{2 + \bar{\gamma}_b} \tag{1}$$

where is $\bar{\gamma}_b$ the average signal-to-noise ratio, defined as

$$\bar{\gamma}_b = \frac{\xi_b}{N_o} E(\alpha^2) \tag{2}$$

The term $E(\alpha^2)$ is the average value of the Rayleigh distributed envelope squared.

When L-th order antenna diversity technique is used, the performance of square-law-detected binary FSK is well approximated by [9]

$$P_2 \approx \left(\frac{1}{\bar{\gamma}_b}\right)^L \binom{2L-1}{L} \tag{3}$$

For dual antenna diversity, L is set equal to two.

A. Performance of Reed-Solomon and BCH Codes

With Reed-Solomon (RS) codes over GF(2^q), q binary channel symbols are used to form one code symbol. Therefore, the probability of code symbol error is $P = 1 - (1 - P_2)^q$, where P_2 is given by (1) and (2), with and without antenna diversity, respectively. For M-ary ($M = 2^q$) block codes with error correction decoding, the decoded symbol error probability can be approximated as

$$P_{es} \approx \frac{1}{N} \sum_{i=t+1}^{N} i \binom{N}{i} P^i (1-P)^{N-i} \tag{4}$$

where $t = [(d - 1)/2]$ is the number of errors that can be corrected by the code, d is the minimum distance of the code, N is the block length, and $[x]$ is the largest integer less than or equal to x. The probability of a decoded bit error is

$$P_{eb} = \frac{2^{q-1}}{2^q - 1} P_{es} \tag{5}$$

For BCH codes, the probability of decoded bit error is just $P_{eb} = P_{es}$, where P_{es} is given by (4). Thus, by combining (4), (5) and (2) or (3), the probability of decoded bit error for block codes can be computed at any signal-to-noise ratio, with or without antenna diversity.

B. Performance of Convolutional Codes

Next, we consider the error rate performance of convolutional codes decoded using the Viterbi algorithm with a hard decision decoding metric. The decoded bit error probability of a rate b / n convolutional code can be upperbounded by

$$P_b < \frac{1}{b} \sum_{l=d}^{\infty} \beta_l D^l \tag{6}$$

where d is the minimum free distance of the code, and β_l is the total information weight of all paths of distance l from the all-zeros path. With hard decision decoding

$$D = \sqrt{4P_2(1-P_2)} \tag{7}$$

where P_2 is given by (1) and (2), with and without antenna diversity, respectively.

The weight and distance structure of many convolutional codes have been computed and tabulated in the literature. Thus, equations (6) and (7) can be used to evaluate the performance of a specific convolutional code.

To study the performance advantage of coded systems over an uncoded one, a system simulation was designed and completed. For decoding Reed-Solomon and BCH codes, the Berlekemp-Massey algorithm and Forney algorithm were implemented. The Viterbi algorithm was employed for decoding convolutional codes.

The simulation results were checked against the corresponding analytical results whenever possible in order to verify the accuracy of our simulation software during the initial phases of the simulation development. It was found that the assumption of ideal interleaving was good. Analytical expressions (5) and (6) produced fairly tight upper bounds. For each coding scheme, the difference between the performance curves obtained from the upper bound and the simulation is less than 1 dB for $10^{-2} < \text{BER} < 10^{-4}$. The simulation results were used to compute coding gains. Fig. 2 illustrates typical BER curves generated from the simulation data for a particular convolutional code with and without dual antenna diversity. Some of the coding gain data are summarized in Table 1. It shows that using the rate-1/2, 32 state (ν=5) convolutional code resulted in near minimum SNR at the target BER, with and without antenna diversity. The coding gain achieved is 19.1 dB with antenna diversity, compared to 13.0 dB with a single antenna. This highly motivates the use of both channel coding and antenna diversity in the system architecture.

The code rate and code size for the codes in Table 1 were chosen based on their potential to meet code performance with decoding delay tolerable for voice transmission, practical code complexity, and channel bandwidth constraints.

To evaluate and compare the delay of the selected codes, we defined three different delay measurements: decoding delay, interleaving delay and overall delay. The delay quantities were specified in signaling interval units, Ts, which provide a normalized delay representation. The definition of decoding delay is the usual delay associated with the decoding algorithm, which has to receive some number of encoded symbols before it can decode them into information bits. The decoding delay is denoted by τ_d. When interleaving is employed along with coding, the overall delay is defined as the sum of the decoding delay and interleaving delay. In other words, $\tau_c = \tau_d + \tau_i$. The interleaving delay, denoted by τ_i, for a (I, J) convolutional interleaver is proportional to (I - 1) × J. (I, J) interleaving produces a minimum separation of at least I for any two code symbols generated by the encoder with a separation of J - 1. The values of I and J are chosen to randomize the error bursts of the fading channel and depend on the coding scheme, the code parameter, and the number of channel symbols per slot (each slot corresponds to a frequency hop).

To show how delay is determined by the system architecture, examine a system utilizing BFSK modulation and a Reed-Solomon code over GF(2^q). The code has a block length spanning $(2^q - 1) \times q$ channel symbols and thus, $\tau_d/Ts = (2^q - 1) \times q$. Now assume there are $c \times q$ channel symbols per slot or equivalently c code symbols per slot. Then in order to ensure that all symbols in a code word suffer from independent fading, successive code symbols must be interleaved across slots. This requires that $I = c$ and $J = (2^q - 1)$. Therefore, $\tau_i/Ts = (c - 1) \times (2^q - 1) \times q$ and $\tau_c/Ts = (2^q - 1) \times c \times q$.

As another example, again consider a BFSK system and assume it has c x q channel symbols per slot. But, in this example the coding scheme is a rate 1/2 convolutional code with memory ν (no. of states = 2^ν). It was shown that the full minimum distances of the 32 the 64 state codes are obtained with a truncation depth of 19 and 27, respectively [10]. These parameter were used in computing the delay stated in Table 1. For the purpose of delay comparison, we assume that the full minimum distance is obtained with a truncation depth of $4.5 \times \nu$. This implies $\tau_d/Ts = 9 \times \nu$. Setting $I = c \times q$ and $J = 9 \times \nu$ as the interleaving parameters guarantees independent fading condition for the successive symbols generated by the encoder with a separation less than truncation depth. Thus, $\tau_i/Ts = (c \times q - 1) \times 9 \times \nu$, and $\tau_c/Ts = 9 \times \nu \times c \times q$.

It can also be shown that using BCH codes with block length N results in $\tau_d/Ts = N$, $\tau_i/Ts = N \times (c \times q - 1)$, and $\tau_c/Ts = N \times c \times q$. Note that under the similar systems assumption, the overall delays for the BCH and Reed-Solomon codes with equal block lengths are the same. The overall delay expressions derived above imply that a Reed-Solomon code over GF(2^q) has comparable overall delay to a convolutional code with memory $\nu = 2^q/9$.

Table 1: Code Performance and Delay Comparison

Code Type RS = Reed-Solomon CC = Convolutional	**Gain w/Single Antenna** (dB)	**Gain w/Dual Antenna** (dB)	**Coding Delay w/o Interleaving** (Ts)	**Overall Coding Delay** (Ts)
BCH (n=31, k=16, d=7)	11.5	17.9	31	310
BCH (n=63, k=36, d=11)	12.3	18.3	63	378
RS (n=31, k=15, d=17)	13.1	18.2	155	310
RS (n=63, k=33, d=31)	13.2	18.7	378	378
CC (r=1/2, v=5)	13.5	19.3	38	380
CC (r=1/2, v=6)	13.6	19.5	54	540

Codes satisfying the voice transmission delay requirement are listed in Table 1. To compute the delay in seconds, we made the assumptions that the delay constraint is 20 ms, each slot consists of 6 BFSK signals for the length 63 block codes and 10 for remaining codes, and the channel transmission rate is 16 kb/s.

To ensure practical code complexity, code selections were limited to block codes with block length less than 255 and convolutional codes with 128 states or less. These are codes which are currently used in various digital communication systems. Channel bandwidth constraints were included in the code selections by limiting the code rate to approximately 1/2; this limits the channel bit rate to a value no larger than twice the user data rate. This limitation was imposed to avoid excessively large bandwidth requirements with high data rates and to decrease link vulnerability from frequency selective fading at high data rates.

Code complexity was quantified in terms of multiplies per decoded symbol for the block codes and additions per decoded symbol for the convolutional code. We found that decoding a single t-error correcting BCH code word required approximately $10t^2 + 3tn$ multiplications, obtained by adding $10t^2$ multiplications for executing the Berlekemp-Massey algorithm to 2tn for evaluating the syndrome, and tn for Chien search. For a t-error correcting RS code, the number of multiplications required to decode a code word increases due to the multiplications in the Forney algorithm. However, the number of operations normalized per bit for RS codes is not necessarily higher than BCH codes when the code length and rate are fixed. In fact, our calculation showed that the (63, 33) RS codes required 26 multiplies per bit versus 33 multiplies per bit for the (63, 36) BCH code. For a 2^v convolutional code, a total of $3*2^v$ addition operations are necessary at each stage. This includes two additions to compute the cumulative path metrics of the paths merging at each state and a comparison (subtraction) to determine which incoming path survives. For convolutional codes, in addition, a trace back operation is required to complete the decoding. For the codes parameters listed in Table 1, the convolutional codes required more than twice the operations per bit than the comparable block codes. This fact was partly confirmed by the longer simulation times for convolutional codes.

The results of analysis and simulation indicate that for a system architecture employing slow frequency hop and dual antenna diversity to combat multipath fading, a BCH code performs as well as an RS code with a comparable code rate. Our investigation also shows that a rate-1/2, 32 state convolutional code can attain a relatively large coding gain at 10^{-3} BER while meeting the tight delay constraint for two-way speech transmission. In addition, convolutional codes offer the advantage of efficient soft decision decoding. The conclusion is that complexity and performance for the code candidates are similar for hard decision decoding; however, the dependence on soft decision decoding must be determined before deciding which code is the best candidate. We now examine some of the relevant tradeoffs.

IV. Decoding Metrics

For the Rayleigh fading channel, soft decision decoding effectively doubles the diversity order available through coding, assuming perfect channel estimates [9]. However, while it is relatively easy to estimate the power of the desired signal for useful signal to interference ratios, it is quite difficult to accurately estimate the interference power. One way to form an estimate of the SNR for a slot is to accumulate the mean squared error between the received signal and hard decision demodulation. The larger the mse, the less reliable the slot. The number of levels of quantization to be used in subsequent decoding depends on the application. For example, with a very slowly changing channel, results could be accumulated over several slots, and many bits of soft decision information extracted. At the other extreme, for only a small number of data symbols per slot the best that can be expected is to be able to declare erasures. The same is surely true of fast hopped systems, where all that could be done is to monitor the received signal strength and declare erasures when it is below some empirically determined threshold.

The estimation of channel parameters from the data and with training sequences was investigated in [5], in the context of DQPSK. It was found that it is better from the point of view of capacity to form estimates directly from the data-bearing signals for short slots of 8-16 symbols, rather than appending a training sequence. In addition, it was found that for some combinations of cell loading and channel codes it was better to use an erasure-declaring mechanism than to use the soft metric proposed; but in any case performance was always better than using simple hard decisions for a properly chosen erasure threshold. Simulations were also conducted using the soft metric with perfect channel knowledge, revealing a very large gap in performance. Thus, channel state information can be very valuable in decoding.

For binary non-coherent FSK, the relevant quantities for soft decision decoding are the received signal power, the ratio of the decision variables for the two frequencies, and the accumulated error for the slot. We are presently investigating several alternative ways of combining this information for slow-hopped systems.

Consider the likelihood function for a communication system involving binary non-coherent FSK, in which one of two frequencies is transmitted with equal probability to a receiver and the transmitted signal is corrupted by additive white Gaussian noise with spectral density $N_o/2$. Then, a model of the received signal is

$$r_i(t) = \alpha \sin(\omega_i t + \phi) + n(t) ; i = 0, 1 \tag{8}$$

We assume that the amplitude, α, is known and the phase, ϕ, is a random variable uniformly distributed in the interval $[0, 2\pi]$. It can be shown that when the constant terms are eliminated, the log-likelihood functions may be written

$$ln(p_i(r)) \sim ln\left(I_o\left(\frac{2\alpha z_i}{N_o}\right)\right); i = 0, 1 \tag{9}$$

where z_i is the decision variable produced by a square law detector (the optimum demodulator for non-coherent FSK), and $I_o()$ is the modified Bessel function [11].

Simulation of a binary non-coherent FSK system using soft decision decoding with the likelihood metric given in (9) was conducted. We assumed that the receiver has perfect knowledge of channel parameters α and N_o. The simulation result showed that soft decision decoding performed better than hard decision decoding, as expected. In particular, a 3.5 dB improvement was obtained with soft decision decoding of a rate-1/2, 32 state convolutional code. When dual antenna diversity was employed, the improvement decreased to approximately 2 dB. Fig. 3 illustrates the improvements in signal-to-noise ratio obtained by using soft decision decoding instead of hard decision decoding on a Rayleigh fading channel.

Given the outcome for the soft decision decoding of convolutional codes, we investigated error and erasure correction decoding for RS codes, which achieves some performance benefit with a trivial increase in decoding computation in comparison to error correction decoding [12]. The mechanism for declaring erasures is based on a ratio threshold test, in which the code symbols having the lowest signal envelope ratio, z_0/z_1 (assuming that $z_0 > z_1$), is erased. When $z_0/z_1 > \theta > 1$, the decision corresponding to z_0 appears to have a good quality. This erasure declaration metric recognizes that the worst BFSK symbol in each q-bit code symbol is the weak link but it will inevitably fail to erase some symbols which are in error, and will erase some symbols which are not in error. There is an optimum range of values for the number of erasures, N_e, declared so that residual error correcting capability, N_c, is sufficient to correct the unerased errors remaining in the received word. The optimum range of values was found during simulation. Simulation of the erasure and error correction decoding for a RS code was carried out and the results are summarized in Table 2. The simulation data showed that a rate-1/2, 32 state convolutional code with soft decision decoding requires approximately 2 dB less signal-to-noise at 10^{-3} BER than a length 31, RS code with erasure and error correc-

Table 2: Code Performance for Different Decoding Methods

Code Type RS = Reed-Solomon CC = Convolutional	**Decoding method**	**Gain w/Single Antenna** (dB)	**Gain w/Dual Antenna** (dB)
RS (n=31, k=15, d=17)	Error Correction	13.1	18.2
	Erasure & Error Correction	14.9	19.5
CC (r=1/2, v=5)	Hard Decision Decoding	13.5	19.3
	Erasure Decoding	15.0	20.0
	Soft Decision Decoding	17.0	21.5

tion decoding on a Rayleigh fading channel, both with and without dual antenna diversity. Thus, a convolutional code with soft decision decoding seems to be the most suitable code for our particular frequency hopped wireless architecture

V. Conclusion

We have considered channel coding options for a frequency-hopped system employing non-coherent binary FSK modulation. We have concluded that the hopping patterns should be selected so as to randomize the interference encountered in successive hops, and have observed the advantages offered by the Latin squares construction for synchronous systems. Alternative channel codes were compared on the basis of delay, complexity, and bandwidth efficiency, with the conclusion that when hard decision decoding is employed, the BCH, RS, and convolutional codes that can meet the delay and bandwidth constraints have similar performance and comparable complexity. At the desired bit error rate, the convolutional codes are slightly better since they required approximately one dB less signal-to-noise ratio than the best performing block codes. Furthermore, the performance gain obtained by soft decision decoding of the convolutional codes was shown to be more significant than the gain obtained by erasure and error correction decoding of block codes. The benefits of dual antenna diversity in combination with coding were demonstrated.

References

[1] J. Min, A. Rofougaran, V. Lin, M. Jensen, A. Abidi, G. Pottie, and Y. Rahmat-Samii, “A Low-Power Handheld Frequency-Hopped Spread Spectrum Transceiver Hardware Architecture,” Virginia Tech Symposium on Wireless Personal Communications, Blackburg, VA, June 1993.

[2] B. Gudmundson, J. Sköld, and J.K. Ugland, “A Comparison of CDMA and TDMA Systems,” Proc. 1992 IEEE Vehic. Tech. Conference, Denver, May 11-13, 1992, pp. 732-735.

[3] H. Sasaoka, “Block coded 16-QAM/TDMA cellular radio system using cyclical slow frequency hopping,” Proc. 1992 IEEE Vehic. Tech. Conference, pp. 405-408.

[4] N.R. Livneh, R. Meridan, M. Ritz, and G. Silbershatz, “Frequency hopping CDMA for digital radio,” Proc. Int’l. Commsphere Symposium, Herzliya, Israel, Dec. 1991.

[5] G.J. Pottie and A.R. Calderbank, “Channel Coding Strategies for Cellular Radio,” IEEE International Symposium on Information Theory, San Antonio, TX, Jan. 1993. (submitted to IEEE Transactions on Vehicular Technology).

[6] A.D. Wyner, “Multi-tone multiple-access for cellular systems,”, AT&T Bell Laboratories Tech. Memo, BL011217-920811-11TM, Nov. 1991.

[7] J. Roche, “Families of binary matrices having nearly orthogonal rows,” AT&T Bell Laboratories Tech. Memo, BL011217-920812-12TM

[8] K.S. Gilhousen, et. al., “on the Capacity of a Cellular CDMA System,” IEEE Trans. Vehic. Tech., vol. 40, May 1991, pp. 303-312.

[9] J.G. Proakis, *Digital Communications.* 2nd Ed. New York: McGraw-Hill, 1989.

[10] S. Lin and D.J. Costello, *Error Control Coding: Fundamentals and Applications.* Englewood Cliffs, NJ: Prentice-Hall, 1983.

[11] A.D. Whalen, *Detection of Signals in Noise.* San Diego: Academic Press, Inc., 1971.

[12] R. E. Blahut, *Theory and Practice of Error Control Codes.* New York: Addison-Wesley, 1983.

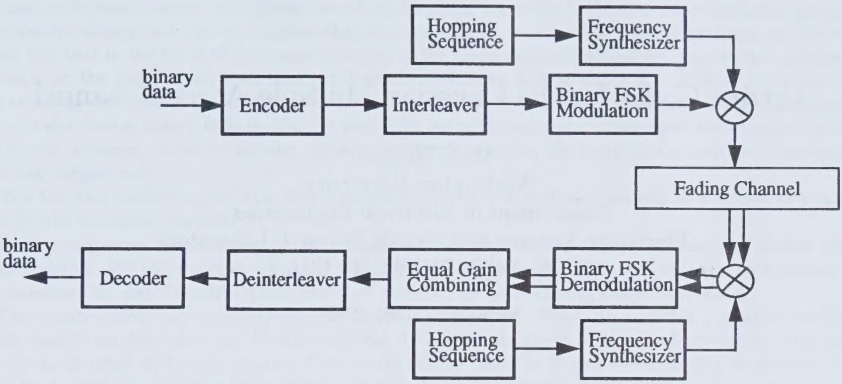

Hopping Sequence
Frequency Synthesizer
binary data
Encoder
Interleaver
Binary FSK Modulation
Fading Channel
binary data
Decoder
Deinterleaver
Equal Gain Combining
Binary FSK Demodulation
Hopping Sequence
Frequency Synthesizer

Fig. 1. System Model.

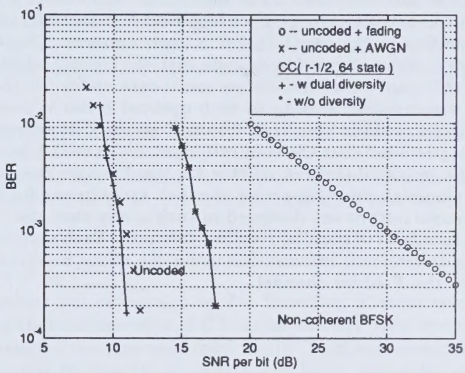


Fig. 2. BER vs. SNR for rate-1/2, 64 state code.

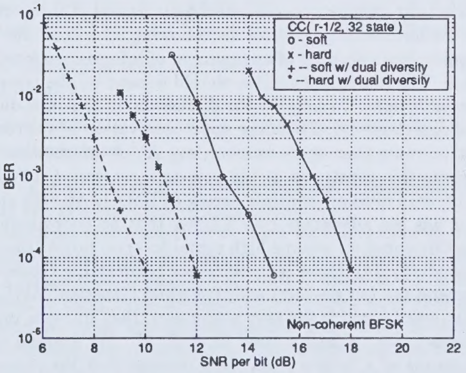


Fig. 3. BER vs. SNR for rate-1/2, 32 state code.

Lattice Codes for the Gaussian Multiple-Access Channel

Bixio Rimoldi*
Washington University
Department of Electrical Engineering
Electronic Systems and Signals Research Laboratory
St. Louis, MO 63130, USA

September 28, 1993

Abstract

Given any group G, the G-adder channel is the channel with T inputs taking values in G and output equal to the sum (over G) of the inputs. A F-adder channel is a G-adder channel where G is the additive group of a finite field F. Similarly, the R-adder channel is the one corresponding to the usual field R of real numbers. The Gaussian multiple-access channel is the R-adder channel in cascade with the (usual) additive white Gaussian noise channel. In this paper multiple-access multi-rate codes for the F-adder channel are defined and a construction leading to such optimal codes is presented. The codes are optimal in that any channel input can transmit at any desired rate and the receiver can decode reliably, provided that the sum rate does not exceed 1 [symbol/channel use]. Note that this is the maximal theoretical rate for the F-adder channel. In order to use such codes on the Gaussian multiple-access channel, transducers for each channel input and one for the channel output are designed in such a way that the resulting channel is equivalent to some number l of parallel F-adder channels. Each such channel can then be used independently to transmit information using the above mentioned multiple-access multi-rate codes for the F-adder channel.

1 Introduction

In this paper we present a new approach to construct multiple-access codes for the Gaussian multiple-access channel [1]. Such a code can be used as a stand-alone multiple-access code when it is guaranteed that no attempt is made to transmit at rates exceeding its capacity, or in conjunction with a random access protocol with multiple reception [2, 3] when the above condition cannot be guaranteed.

Good codes should take into consideration the stochastic nature of the source and the channel [4]. In developing our approach we have been guided by the following considerations: (1) The code assigned to a channel input node should allow encoding of any number of information symbols (or packets) from zero to some maximum number. The maximum number should depend on the power constraint of the specific node. We call such codes *multi-rate*. To simplify the discussion, in this introduction we assume that each node has the same power constraint. (2) Decoding upon observation of the received sum-word should be successful, provided that no more than some number, say K, of information symbols were transmitted among all nodes. We denote this constraint as the *sum-rate constraint.* (3) The maximum overall rate should be at least that of time division multiple access (TDMA). At high signal-to-noise ratios this is almost optimal [4] and it has the advantage over TDMA that no coordination is needed to assign the channel to busy users. In this paper we assume high signal-to-noise ratios (the case for low signal-to-noise ratios is presently being studied). (4) There is no difference between the Gaussian multiple-access channel with only one active user and the (single-user) additive white Gaussian (AWGN) channel. If we want to approach channel capacity when user i is the only active user, then his code must be a "good" single-user code for the AWGN channel, e.g., a lattice code [5, 6, 7, 8]. (5) Assume that user i is transmitting at his maximal rate by means of a lattice code. Now assume that the same user wants to reduce the transmission rate. It seems reasonable to require that this be done by using a subcode of the original code. This choice seems reasonable in order to keep the encoding/decoding complexity manageable (a

---

*Supported by National Science Foundation grant NCR-9109944.

different code would require a different encoding/decoding algorithm). On the other hand, the guidance provided by information theory suggests that there should be no loss of optimality in doing so. We refer to the fact that in the proof of the capacity region of the Gaussian multiple-access channel the rate has no influence on the distribution from which codewords have to be drawn, which is $\sim \mathcal{N}(0, P^i)$ (see e.g. [1]). The rate only determines the number of codewords that need to be selected according to this distribution. Hence, information theory tells us that, in principle, we could construct the largest code needed by user i and then construct codes for smaller rates by randomly selecting the appropriate number of codewords from this largest code.

The last two points suggest that the multiple-access code of each user should be a subset of a lattice code for the Gaussian channel.

In this paper we describe a procedure leading to multiple-access codes that have all of the above characteristics. We will see that when the power constraint is uneven, then the throughput achieved via our codes can be significantly higher than that achieved by time-sharing single-user codes.

Our construction has emerged from the following two steps. First, we consider a simpler multiple-access channel model, called the F-adder channel. Given a finite group F, the F-adder channel associated to it is the channel with some number T of inputs taking values in F and whose output is the sum (over F) of the inputs. The reasons for starting with this channel model are that (i) the multiple-access coding problem is simpler since there is no noise and (ii) there is an extensive theory of error control codes over finite fields F. It turns out that in this simplified setting we can construct powerful codes that have the above properties (1)-(3). The basic idea is that the transposed parity check matrix of a maximum distance separable *error-control* code over F can be used to generate multi-rate multiple-access codes with sum-rate constraint for the F-adder channel. This is done in Section 2. In Section 3 we show how to map codes for the F-adder channel into lattice codes for the Gaussian multiple-access channel.

2 Multiple-Access Codes for F-Adder Channels

We assume that F is a finite field and that there are T users. Unless otherwise specified all additions and multiplications in this section will be in F. The F-adder channel is then the mapping

$$F\Sigma : F^T \to F \tag{1}$$

that sends $c^1, \cdots, c^T$, $c^i \in F$, to $\sum_{i=1}^{T} c^i$.

The set of natural numbers will be denoted by $\boldsymbol{N}$. Hereafter, a *linear single-user code* $\mathcal{C}$ is a linear subspace of F^N. Following standard notation, if $\mathcal{C}$ has dimension K then we will refer to it as an $[N, K]$ code. If $K = 0$ then $\mathcal{C}$ consists of the zero word only. An $[N, K, d]$ code is an $[N, K]$ code with minimum distance d. A *generator matrix* $\boldsymbol{B}$ of an $[N, K]$ code is a K by N matrix for which the rows are a basis for $\mathcal{C}$. Given a K by N matrix $\boldsymbol{B}$, $\boldsymbol{B}_k$, k a positive integer not exceeding K, will be used for the matrix consisting of the first k rows of $\boldsymbol{B}$. The blocklength N and the number of input nodes T will be understood to be known at all times. Input nodes are indexed by superscripts taken from $\mathcal{T} := \{1, \cdots, T\}$.

Definition 1 A *multiple-access code* (for the F-Adder channel), denoted $[N, \boldsymbol{K}]$ code, $\boldsymbol{K} = (K^1, \cdots, K^T) \in \boldsymbol{N}^T$, is an ordered collection $\underline{\mathcal{C}} = \mathcal{C}^1 \times \cdots \times \mathcal{C}^T$ such that $\mathcal{C}^i$ is an $[N, K^i]$ code.

The subspaces $\mathcal{C}^1, \cdots, \mathcal{C}^T$ are *independent* if no sum $\boldsymbol{c}^1 + \cdots + \boldsymbol{c}^T$ with $\boldsymbol{c}^i \in \mathcal{C}^i$ is zero, except for the trivial sum in which $\boldsymbol{c}^i = 0$ for all i.

Definition 2 An *independent multiple-access code* is a multiple-access code $\underline{\mathcal{C}} = \mathcal{C}^1 \times \cdots \times \mathcal{C}^T$ such that $\mathcal{C}^1, \cdots, \mathcal{C}^T$ are independent subspaces of F^N.

If $\boldsymbol{B}^i$ is a generator of $\mathcal{C}^i$, $\underline{\mathcal{C}} = \mathcal{C}^1 \times \cdots \times \mathcal{C}^T$ is an independent multiple-access code if and only if the matrix obtained by stacking the $\boldsymbol{B}^i$s has linearly independent rows.

If node i transmits an arbitrary codeword $\boldsymbol{c}^i$ of $\mathcal{C}^i$, then we can recover $\boldsymbol{c}^i$, from $\sum_{j=1}^{T} \boldsymbol{c}^i$ if and only if $\underline{\mathcal{C}}$ is an independent multiple-access code. So what we need are independent multiple-access codes and we need several of them. A single independent $[N, \boldsymbol{K}]$ code $\underline{\mathcal{C}}$ where K^i/N is the maximal rate needed by node i would be equivalent to using time-division multiple-access with *fixed* time slots. Our approach is to allow each node to select the most appropriate rate for that *frame*. In this section we assume that the receiver knows the transmission rate of each node. This assumption can be relaxed.

For a node transmitting at the rate k/N by means of an $[N, k]$ code $\mathcal{C}$, the natural way to decrease its rate is to choose a subcode of $\mathcal{C}^i$. In order to avoid the reservation process associated with time-division multiple-access, it is important that the code of each node is a function of the rate only and not of the code selected by other nodes.

Definition 3 A *single-node multi-rate code* , denoted (N, K) code, is a collection $\mathcal{D} := \{\mathcal{C}_k : \mathcal{C}_k$ is an $[N, k]$ code, $k = 0, 1, \cdots, K$ and $\mathcal{C}_k \supset \mathcal{C}_j$ if $k > j\}$. The *operational region* of an (N, K) code is the set $\mathcal{K} := \{0, 1, \cdots, K\}$.

Definition 4 A *multiple-access multi-rate code,* denoted $(N, \boldsymbol{K})$ code, is a collection $\underline{\mathcal{D}} := \mathcal{D}^1 \times \cdots \times \mathcal{D}^T$ of multiple-access codes, where for each $i = 1, \cdots, T$, $\mathcal{D}^i$ is an (N, K^i) code.

Note our convention: brackets are used for single-rate codes whereas parentheses are used for multi-rate codes. Hence, the four possibilities $[N, K]$, (N, K), $[N, \boldsymbol{K}]$, and $(N, \boldsymbol{K})$ uniquely specify the type of code, i.e., single-node single-rate, single-node multi-rate, multiple-access single-rate, and multiple-access multi-rate, respectively.

For every $(N, \boldsymbol{K})$ code $\underline{\mathcal{D}}$ and every *configuration* $\boldsymbol{k} \in \mathcal{K}^1 \times \cdots \times \mathcal{K}^T$, there is exactly one $[N, \boldsymbol{k}]$ code $\underline{\mathcal{C}}$ in $\underline{\mathcal{D}}$. This multiple-access code will be denoted by $\underline{\mathcal{D}}(\boldsymbol{k})$.

Definition 5 Let $\underline{\mathcal{D}}$ be an $(N, \boldsymbol{K})$ code. Its *operational region* is the set $\underline{\mathcal{K}} := \{\boldsymbol{k} \in \mathcal{K}^1 \times \cdots \times \mathcal{K}^T : \underline{\mathcal{D}}(\boldsymbol{k})$ is an independent $[N, \boldsymbol{k}]$ code $\}$.
Moreover, an $(N, \boldsymbol{K})$ code with operational region $\underline{\mathcal{K}}$ will be referred to as an $(N, \underline{\mathcal{K}})$ code.

Given an (N, K) code $\mathcal{D} = \{\mathcal{C}_0, \mathcal{C}_1, \cdots, \mathcal{C}_K\}$, it is always possible to choose a basis of $\mathcal{C}_j$, $j = 1, \cdots, K$ by extending a basis of $\mathcal{C}_{j-1}$ by one coordinate. Hence, one can describe an (N, K) code by a generator $\boldsymbol{B}$ organized in such a way that the first row is a basis for $\mathcal{C}_1$, the first two rows are a basis for $\mathcal{C}_2$, ...and so on. One can make the matrix $\boldsymbol{B}$ unique by requiring that the first nonzero element of each row be one. Hereafter the generator matrix of an (N, K) code will be this matrix. As for block codes, the generator $\boldsymbol{B}$ of an $(N, \boldsymbol{K})$ code $\mathcal{D}$ describes an encoder that maps the information sequence $\boldsymbol{a} := (a_1, \cdots a_K)$ to the codeword $\boldsymbol{aB}$. According to our basis convention, we use the single-node multi-rate code $\mathcal{D}$ at the rate k/N, $k \in \{0, 1, \cdots, K\}$, by forcing the last $(K - k)$ positions of $\boldsymbol{a}$ to be zero. Similarly, an $(N, \boldsymbol{K})$ code $\underline{\mathcal{D}} = \mathcal{D}^1 \times \cdots \times \mathcal{D}^T$ can be described by T generator matrices $\boldsymbol{B}^1, \cdots, \boldsymbol{B}^T$, where $\boldsymbol{B}^i$ is a K^i by N matrix whose first k rows, $k \leq K^i$, form a basis for the $[N, k]$ code in $\mathcal{D}^i$.

In a scheme with many nodes, the maximal rate K^i/N of an arbitrary node i is typically much smaller than 1. Let $\boldsymbol{K} = (K^1, \cdots, K^T)$ be the T-tuple describing the maximal rate needed by each user. Given $\boldsymbol{K}$, it is desirable to have an $(N, \boldsymbol{K})$ code with operational region containing

$$\{\boldsymbol{k} \in \mathcal{K}^1 \times \cdots \times \mathcal{K}^T : \sum k^i \leq K\}, \tag{2}$$

where K is some positive integer not exceeding N.

Definition 6 An $(N, \boldsymbol{K}, K)$ code is an $(N, \boldsymbol{K})$ code whose operational region contains (2).

Definition 7 An $(N, \boldsymbol{K})$ code is *optimal* if it is an $(N, \boldsymbol{K}, K)$ code with $K = N$.

Now we describe a construction leading to $(N, \boldsymbol{K}, K)$ codes.

Code Construction A

Let

$$H^T = \begin{pmatrix} \boldsymbol{h}_1 \\ \vdots \\ \boldsymbol{h}_n \end{pmatrix}, \ \boldsymbol{h}_i \in F^{n-k},$$

be the transposed parity check matrix of an $[n, k, d]$ code over F and let $\boldsymbol{K} \in \boldsymbol{N}^T$ be such that $K^i \leq d-1$ for all $i \in \mathcal{T}$ and $\sum K^i = n$. It is well known that any $d-1$ rows of H^T are linearly independent. Partition H^T into T matrices

$$H^T = \begin{pmatrix} \boldsymbol{B}^1 \\ \vdots \\ \boldsymbol{B}^T \end{pmatrix},$$

in such a way that $\boldsymbol{B}^i$ has K^i rows. Each $\boldsymbol{B}^i$ is the generator matrix of an (N, K^i) code $\mathcal{D}^i$ since it has linearly independent rows. The code of construction A is the $(N, \boldsymbol{K})$ code $\underline{\mathcal{D}} := \mathcal{D}^1 \times \cdots \times \mathcal{D}^T$. □

Theorem 1 The $(N, \boldsymbol{K})$ code of construction A is an $(N, \boldsymbol{K}, K = d - 1)$ code.

Proof. We have to show that if $\boldsymbol{k} \in \mathcal{K}^1 \times \cdots \times \mathcal{K}^T$ is such that $\sum k^i \leq d-1$ then $\mathcal{C}^1 \times \cdots \times \mathcal{C}^T := \underline{\mathcal{D}}(\boldsymbol{k})$ is an independent multiple-access code. $\mathcal{C}^i$ has generator matrix $\boldsymbol{B}^i_{k^i}$ and the matrix

$$\begin{pmatrix} \boldsymbol{B}^1_{k^1} \\ \vdots \\ \boldsymbol{B}^T_{k^T} \end{pmatrix}$$

consists of $\sum k^i$ distinct rows of H^T; these rows must be linearly independent since $\sum k^i \leq d-1$. □

Example 1 A simple example is obtained by applying construction A to the $[n = 7, k = 4, d = 3]$ Hamming code. A parity check matrix for this code is

$$H = \begin{pmatrix} 1001011 \\ 0101101 \\ 0010111 \end{pmatrix}.$$

Let B^i consist of the i-th row of H^T, $i = 1, \cdots, 7$. Then $\mathcal{D}^i$ is an $(N = 3, K = 1)$ code and $\underline{\mathcal{D}} := \mathcal{D}^1 \times \cdots \times \mathcal{D}^7$ is an $(N = 3, \boldsymbol{K} = (1,1,1,1,1,1,1), K = 2)$ code.

Theorem 2 Whenever construction A operates on an $[n, k, d]$ maximum distance separable (MDS) code, the resulting $(N, \boldsymbol{K})$ code is optimal.

Proof. By definition, an MDS code is an $[n, k, d]$ code that satisfies with equality the Singleton bound; i.e., $n - k \geq d - 1$. From $N = n - k$, $K = d - 1$, and equality in the Singleton bound we obtain $K = N$. Hence, the code obtained via construction A is an $(N, \boldsymbol{K}, K = N)$ code. □

Hence, Reed Solomon (RS) codes are very attractive block codes to use in construction A.

3 Multiple-Access Codes for the Gaussian Channel

Here we show how to map multiple-access codes for F-adder channels into lattice codes. It is convenient to consider first the case without power constraints.

3.1 Codes Without Power Constraints

Hereafter S is an arbitrary lattice, scaled so that the resulting probability of error on the given (single-user) additive white Gaussian noise channel is negligible. Of course, what negligible means depends on the specific application. With the S-adder channel we refer to the Gaussian multiple-access channel with inputs that are elements of the lattice S and a decoder at the output that maps the received point into the nearest lattice point. Our assumption about S implies that the decoder output with and without noise is the same with high probability. Hence we may assume that the S-adder channel is error-free. Hereafter G is its additive group of the finite field F. The relation between the finite field F and the lattice S will be clarified later.

Consider first two users. If, for some positive integer l, it were possible to specify an input transducer

$$M : F^l \to S \tag{3}$$

and an output transducer

$$D : S \to F^l \tag{4}$$

in such a way that for any c^1, $c^2 \in F^l$

$$D(M(c^1) + M(c^2)) = c^1 + c^2, \tag{5}$$

then we could appply the results of Section 2 since the above equation says that the cascade of the two input transducers with the S-adder channel and the output transducer is just equivalent to l parallel F-adder channels. (Alternatively, we can think of l uses of the same F-adder channel but this view will turn out to be less interesting). We could then use the results of Section 2 to each one of the l F-adder channels. This is in principle what we are going to do. We say in principle since it will turn out it is impossible to define input/output transducers that satisfy (5) for all c^1, $c^2 \in F^l$. Some restrictions will be needed, but these restrictions will not affect the result of our strategy.

The mapping performed by the S-adder channel on its inputs will be denoted by $S\Sigma$, i.e.,

$$S\Sigma: \quad \begin{array}{ccc} S^T & \longrightarrow & S \\ x^1, \cdots, x^T & \rightsquigarrow & \sum x^i. \end{array} \tag{6}$$

The l inputs to an input transducer will be written as a column vector

$$\check{c} := \begin{pmatrix} c_0 \\ \vdots \\ c_{l-1} \end{pmatrix}$$

with possible superscript i, $i = 1, \cdots, T$, to denote that it belongs to the input node i. The same symbol G^l will be used to denote the set of l tuples written as rows and the set of l tuples written as columns. Hence, $\check{c}^i \in G^l$ and the modulator is a mapping

$$M : G^l \longrightarrow S \text{ with } M(0) = 0,$$

where the condition $M(0) = 0$ is necessary to account for the fact that users that are not transmitting have their equipment possibly shut off. The output transducer is a mapping

$$D : S \longrightarrow G^l.$$

Now we proceed to define the modulator. Let

$$S_0 := S \supseteq S_1 \supseteq \cdots \supseteq S_l \supseteq S_{l+1} := \{0\},$$

be a chain of sublattices. Since a lattice is a normal group, the quotient groups S_j/S_{j+1}, $j = 0, 1, \cdots, l$ are well defined. We ask that for $j = 0, 1, \cdots, l-1$ the quotient groups S_j/S_{j+1} be finite. For ease of notation we also assume that these quotient groups are all isomorphic to the same finite group G, i.e.,

$$S_j/S_{j+1} \simeq G,\ j = 0, 1, \cdots, l-1.$$

Let $\pi_j : S_j \to S_j/S_{j+1}$, $j = 1, \cdots, l$ be the standard mapping. Recall that π_j is a homomorphism with Kern S_{j+1}. Choose a system of coset representatives $[S_j/S_{j+1}]$. This system induces a 1-1 correspondence $\xi_j : S_j/S_{j+1} \to [S_j/S_{j+1}]$, sending a coset $\bar{x}$ of S_{j+1} in S_j to its coset representative $\xi_j(\bar{x})$. Denote the isomorphism between S_j/S_{j+1} and G by $\varphi_j : S_j/S_{j+1} \to G$.

Let $\rho_j := \xi_j \circ \varphi_j^{-1} : G \to [S_j/S_{j+1}]$, $j = 1, \cdots, l$. Finally, let $\phi_j := \varphi_j \circ \pi_j : S_j \to G$, $j = 1, \cdots, l$. ϕ_j is a homomorphism with Kern S_{j+1}.

We obtain the following diagram:

$$\begin{tikzcd} S_j \arrow[r, "\phi_j", "\text{Hom.}"'] & G \arrow[r, "\rho_j", "\text{1-1}"'] & {[S_j/S_{j+1}]} \end{tikzcd}$$

Every $x \in S$ can be written in a unique way as

$$x = x_0 + x_1 + \cdots + x_{l-1} + t$$

where $x_j \in [S_j/S_{j+1}]$ and $t \in S_l$. Moreover, the 1-1 correspondences ρ_j induce a 1-1 correspondence

$$\tilde{M}: \quad \begin{array}{ccc} G^l \times S_l & \longrightarrow & S \\ (\check{c}, x_l) & \rightsquigarrow & \sum_{j=0}^{l-1} \rho_j(c_j) + x_l. \end{array} \tag{7}$$

Hence, every $x \in S_l$ has a unique representation as

$$x = \rho_0(c_0) + \rho_1(c_1) + \cdots + \rho_{l-1}(c_{l-1}) + x_l,\ c_j \in G \text{ and } x_l \in S_l.$$

We define the input transducer M as the restriction of $\tilde{M}$ to G^l, i.e.,

$$M: \quad \begin{array}{ccc} G_l & \longrightarrow & M(G^l) \\ \check{c} & \rightsquigarrow & \sum_{j=0}^{l-1} \rho_j(c_j). \end{array} \tag{8}$$

Since $\tilde{M}$ is a one-to-one correspondence, its inverse is well defined. One way to specify it explicitly is via $l+1$ mappings p_j, $j = 0, 1, \cdots, l$, where for $j = 0, 1, \cdots, l-1$, $p_j : S_0 \longrightarrow G$ maps $x = \sum_{j=1}^{l} \rho_j(c_j) + x_l$ to c_j and $p_l : S_0 \longrightarrow S_l$ maps x to x_l. These $l+1$ mappings can be defined recursively by

$$\begin{gathered} p_0(x) := \phi_0(x) \\ p_i(x) := \phi_j(x - \textstyle\sum_{j=0}^{i-1} \rho_j(c_j)), i = 1, \cdots, l-1 \\ p_l(x) := x - \textstyle\sum_{j=0}^{l-1} \rho_j(c_j). \end{gathered} \tag{9}$$

Similarly, M is a one-to-one correspondence from G^l onto $M(G^l)$. Its inverse is uniquely determined by the restriction of p_j to $M(G^l)$, $j = 0, \cdots, l-1$. Notice that the set $M(G^l)$ is not closed under addition. In order to make the inverse of M a possible candidate for D, we need to extend it to S_0. A possible extension is obtained by letting p_j be equal to ϕ_j over the entire lattice S_0. This definition of D, however, would not satisfy condition (5). To see this, consider first $p_0 = \phi_0 : S_0 \to G$. If $x' = \sum_{j=0}^{l-1} \rho_j(cj') + x'_l$ and $x'' = \sum_{j=0}^{l-1} \rho_j(c''_j) + x''_l$, then letting $x = x' + x''$ and using the fact that ϕ_0 is a homomorphism, it follows that $p_0(x) = \phi_0(x) = \phi_0(x' + x'') = \phi_0(x') + \phi_0(x'') = p_0(x') + p_0(x'')$. So far so good. Unfortunately things are not so simple for $p_j, j = 1, \cdots, l-1$. To see this, consider $p_1 : S_0 \to G$ sending $x \rightsquigarrow \phi_1(x - \rho_0(c_0))$. This is a homomorphism from S_1 into G but is *not* a homomorphism from S_0 into G since

$$p_1(x') + p_1(x'') := \phi_1(x' - \rho_0(c'_0)) + \phi_1(x'' - \rho_0(c''_0)) = \phi_1(x - (\rho_0(c'_0) + \rho_0(c''_0))) \tag{10}$$

in general differs from

$$p_1(x) := \phi_1(x - (\rho_0(c'_0 + c''_0)) \tag{11}$$

because ρ_0 is *not* a homomorphism. This shows that we have to redefine $p_1 : S_0 \longrightarrow G$ according to the right hand side of (10). But in order to do so we need c'_0 and c''_0 whereas what we have is $c_0 = c'_0 + c''_0$.

If $\boldsymbol{c}'_0$ and $\boldsymbol{c}''_0$ are codewords of a multiple-access code for some F-adder channel as described in Section 2, then from $\boldsymbol{c}_0 = \boldsymbol{c}'_0 + \boldsymbol{c}''_0$ we can determine $\boldsymbol{c}'_0$ and $\boldsymbol{c}''_0$. Hence, we assume that G is the additive group of a finite field F and that for each level j we have an independent $[N, \boldsymbol{K}_j]$ multiple-access code $\underline{\mathcal{C}}_j = (\mathcal{C}_j^1, \cdots, \mathcal{C}_j^T)$ over F and that $\boldsymbol{c}_j^i \in \mathcal{C}_j^i$. Since its inputs are codewords, it is natural to look at the modulator as a mapping that operates on N-length sequences. In general, we will use boldface letters to denote N fold cartesian products of sets such as $\boldsymbol{G}$ and $\boldsymbol{S}$, N tuples such as $\boldsymbol{x}$, and functions operating on them like

$$\begin{array}{rccc} \boldsymbol{S\Sigma}: & \boldsymbol{S}^T & \longrightarrow & \boldsymbol{S} \\ & \boldsymbol{x}^1, \cdots, \boldsymbol{x}^T & \rightsquigarrow & \sum \boldsymbol{x}^i. \end{array}$$

An input to the modulator $\boldsymbol{M}$ is then an $l \times N$ matrix that will be denoted by C. Now we may define $\tilde{\boldsymbol{D}} : \boldsymbol{S}_0 \longrightarrow \boldsymbol{G}^l \times \boldsymbol{S}_l$ to have the desired property. The new definition depends on the independent multiple-access codes $\underline{\mathcal{C}}_j = (\mathcal{C}_j^1, \cdots, \mathcal{C}_j^T)$, $j = 0, \cdots, l-1$. The definition of $\tilde{\boldsymbol{D}}$ is given in terms of the algorithm below. `Decoder`$_j$: $\boldsymbol{G} \to \boldsymbol{G}^T$ is *any* extension to $\boldsymbol{G}$ of the function that maps $\sum_{i=1}^{T} \boldsymbol{c}_j^i$, $\boldsymbol{c}_j^i \in \mathcal{C}_j^i$, to $\boldsymbol{c}_j^1, \cdots, \boldsymbol{c}_j^T$. We need such an extension to guarantee that $\tilde{\boldsymbol{D}}$ is defined also for those received lattice points $\boldsymbol{x} \in \boldsymbol{S}$ that did not originate from using the modulator inputs to transmit multiple-access codewords. When such points are received, however, a reliable decoding operation cannot be guaranteed.

Algorithm specifying $\tilde{\boldsymbol{D}}$

```
Begin
Initialization
  $j := 0$
  Read $\boldsymbol{x}$                    (Note 1: $\boldsymbol{x} \in \boldsymbol{S}_j$)
Loop
  $\boldsymbol{c}_j := \boldsymbol{\phi}_j(\boldsymbol{x})$                (Note 2: $\boldsymbol{\phi}_j : \boldsymbol{S}_j \to \boldsymbol{G}$ is a homom.)
  $(\boldsymbol{c}_j^1, \cdots, \boldsymbol{c}_j^T)$ := Decode$_j(\boldsymbol{c}_j)$
  $\boldsymbol{x} := \boldsymbol{x} - \sum_{i=1}^{T} \rho_j(\boldsymbol{c}_j^i)$        (Note 3: $\boldsymbol{x} \in \boldsymbol{S}_{j+1}$)
  $j := j + 1$
Repeat until $j = l$
Output $C := \begin{pmatrix} c_0 \\ \vdots \\ c_{l-1} \end{pmatrix}$
Output $\boldsymbol{x}_l := \boldsymbol{x}$
End.
```

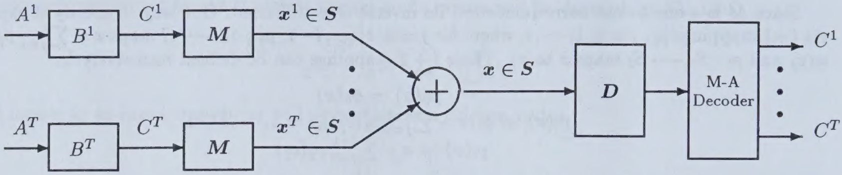

A^1
B^1
C^1
M
$\boldsymbol{x}^1 \in \boldsymbol{S}$
A^T
B^T
C^T
M
$\boldsymbol{x}^T \in \boldsymbol{S}$
$+$
$\boldsymbol{x} \in \boldsymbol{S}$
$\boldsymbol{D}$
M-A
Decoder
C^1
C^T

Figure 1: Overall system.

Notice that the above algorithm in effect internally determines the input of each modulator C^i, $i = 1, \cdots, T$. Hence, $\tilde{\boldsymbol{D}}$ actually incorporates a demodulator and a decoder. For conceptual reasons we like to think of these two operations as being done separately.

Example 2 Let $S = S_0 = \mathsf{N}$, the integer lattice. $\mathsf{N} \subset \mathsf{R}$ so that the lattice dimension $n = 1$. Let $S_1 = 7S_0 = 7\mathsf{N}$ (the sublattice whose elements are integer multiples of 7), and let $S_2 = 7S_1 = 7^2\mathsf{N}$. Then S_i/S_{i+1}, $i = 0, 1$, is isomorphic to the additive group G of the Galois field F_7 that can be represented by the integers $\{0, 1, \cdots, 6\}$ with addition and multiplication modulo 7. Let $[S_0/S_1] = \{0, 1, 2, 3, -3, -2, -1\}$ and $[S_1/S_2] = \{0, 7, 14, 21, -21, -14, -7\}$. Let $\rho_0 : F_7 \longrightarrow [S_0/S_1]$ send a 7-ary integer g_0 to the unique integer x_0 in $[S_0/S_1]$ which is congruent to g_0 modulo 7. Similarly, let $\rho_1 : F_7 \longrightarrow [S_1/S_2]$ send a 7-ary integer g_1 to the unique integer x_1 in $[S_1/S_2]$ which is congruent to $7g_1$ modulo 7^2. This completely specifies $\tilde{M}$

$$\begin{array}{rccc} \tilde{M}: & F_7 \times F_7 \times S_2 & \longrightarrow & S_0, \\ & (g_0, g_1, x_3) & \rightsquigarrow & \rho_0(g_0) + \rho_1(g_1) + x_3 \end{array}$$

as well as

$$\begin{array}{rccc} M: & F_7 \times F_7 & \longrightarrow & S_0. \\ & (g_0, g_1) & \rightsquigarrow & \rho_0(g_0) + \rho_1(g_1) \end{array}$$

Now we need to define a multiple-access code $\underline{\mathcal{D}}$. A convenient way to do so is via code construction A, starting with a Reed-Solomon (RS) code over some extension field F_{7^m} of F_7. Since RS codes are maximum distance separable (MDS) codes, $\underline{\mathcal{D}}$ will then be optimal as defined in Section 2. The size of the subfield determines the largest codeword length of the RS code that can be constructed over that field. In turn, it determines the maximum value of $\sum K^i$. For simplicity we choose $m = 1$ and start with a $[6, 3, 4]$ RS code over F_7. For definiteness, we construct it via the primitive 7-th root of unity $\alpha = 3$ in F_7 and generator polynomial $g(x) = (x - 1)(x - \alpha)(x - \alpha^2)$. A transposed parity check matrix for this code is

$$H^T = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 3 & 2 \\ 1 & 2 & 4 \\ 1 & 6 & 1 \\ 1 & 4 & 2 \\ 1 & 5 & 4 \end{pmatrix}.$$

According to construction A of Section 2, we assign to node 1 the generator matrix consisting of the first two rows of H^T, to node 2 the generator matrix consisting of rows 3 and 4 of H^T, and to node 3 the generator matrix consisting of the last two rows of H^T. The resulting generator matrices can actually be augmented by one row each using a modified version of construction A called construction B (omitted here for space reasons). The result is

$$B^1 = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 3 & 2 \\ 1 & 6 & 1 \end{pmatrix}, B^2 = \begin{pmatrix} 1 & 2 & 4 \\ 1 & 6 & 1 \\ 1 & 5 & 4 \end{pmatrix}, B^3 = \begin{pmatrix} 1 & 4 & 2 \\ 1 & 5 & 4 \\ 1 & 3 & 2 \end{pmatrix},$$

respectively. This defines a $(N = 3, \boldsymbol{K} = (3, 3, 3), K = 3)$ optimal multiple-access code for the F_7-adder channel. This code can be used as shown in Fig. 1.

A set of valid information matrices is:

$$A^1 = \begin{pmatrix} 5 & 0 & 0 \\ 1 & 4 & 0 \end{pmatrix}, A^2 = \begin{pmatrix} 4 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, A^3 = \begin{pmatrix} 6 & 0 & 0 \\ 2 & 0 & 0 \end{pmatrix}.$$

ΞΞΞThose information matrices can be recovered at the receiver since for each row, the total number of information symbols taken over all matrices does not exceed (in this case is equal to) $K = 3$. The i-th modulator input is then $C^i = A^i B^i$, i.e.,

$$C^1 = \begin{pmatrix} 5 & 5 & 5 \\ 5 & 6 & 2 \end{pmatrix}, C^2 = \begin{pmatrix} 4 & 1 & 2 \\ 0 & 0 & 0 \end{pmatrix}, C^3 = \begin{pmatrix} 6 & 3 & 5 \\ 2 & 1 & 4 \end{pmatrix}.$$

The modulator output sequences are $\boldsymbol{x}^i = \rho_0(\boldsymbol{c}_0^i) + \rho_1(\boldsymbol{c}_1^i)$, i.e., $\boldsymbol{x}^1 = (-16, -9, 12)$, $\boldsymbol{x}^2 = (-3, 1, 2)$, and $\boldsymbol{x}^3 = (13, 10, -23)$, and the received sequence is $\boldsymbol{x} = \sum \boldsymbol{x}^i = (-6, 2, -9)$. The demodulator first finds $\rho_0(\boldsymbol{c}_0)$ by mapping $\boldsymbol{x}$ into $[\boldsymbol{S}_0/\boldsymbol{S}_1]$. This yields $\rho_0(\boldsymbol{c}_0) = (1, 2, 2)$. Then it finds $\boldsymbol{c}_0 = \rho^{-1}(1, 2, 2) = (1, 2, 5)$. In this example we assume that the decoder knows the the number of (active) information symbols in each row of each user. Then the three information symbols of level 0 can be found as the only solution to the equation

$$(a_1^1, a_1^2, a_1^3) \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 4 & 2 \end{pmatrix} = (1, 2, 5),$$

which is $(5, 4, 6)$. Now the decoder can determine the first row $\boldsymbol{c}_0^i$ of each information matrix. Hence, it can determine $\boldsymbol{x} - (\rho_0(\boldsymbol{c}_0^1) + \rho_0(\boldsymbol{c}_0^2) + \rho_0(\boldsymbol{c}_0^3))$ which is $(0, 0, -7)$. This is $\rho_1(\boldsymbol{c}_1)$. From here it can determine $\boldsymbol{c}_1 = (0, 0, 6)$ and finally find the information symbols at level 1 by solving the equation

$$(a_2^1, a_2^1, a_2^3) \begin{pmatrix} 1 & 1 & 1 \\ 1 & 3 & 2 \\ 1 & 4 & 2 \end{pmatrix} = (0, 0, 6).$$

The solution is $(1, 4, 2)$ which is the correct result. At this point the information matrices A^1, A^2, and A^3 are completely known to the decoder.

3.2 Codes With Power Constraint

Here we extend the results of the previous subsection to include power constraints. For simplicity we will not proceed with the most general case. Instead, we assume that $S_{j+1} = pS_j$ for some prime number p, $j = 0, \cdots, l-1$ and that each level uses a copy of the same $(N, \boldsymbol{K}, N)$ optimal multiple-access code $\underline{\mathcal{D}}$ overF_p.

The power constraint is described by the T-tuple $\boldsymbol{P} = (P^1, \cdots, P^T)$. The power constraint P^i of node i is enforced by a bounding region $\mathcal{R}(P^i)$ which is obtained by appropriately scaling some bounding region $\mathcal{R}$ which is the same for all nodes. Hence, the largest code of node i becomes $S \cap \mathcal{R}(P^i)$. Codes of this type defined via a lattice and a bounding region are called *lattice codes.* In principle, one should choose the bounding region in R^n that best approximates an n sphere. In practice, however, one has to be concerned with the difficulty of mapping information symbols to codewords and codewords back to information symbols. Conway and Sloane [5] have found an elegant way to do so, together with fast algorithms to do all the necessary operations. The idea is to require that the bounding region be the Voronoi region of some sublattice.

Even if we didn't say so, this idea came up naturally in example 2. In fact, we chose the coset leaders $[S_0/S_1]$ in such a way as to minimize energy. The result was $[S_0/S_1] = S_0 \cap V_1$ where V_1 denotes the Voronoi region of S_1. Similarly, we chose the coset leaders of $[S_1/S_2]$ in such a way that $[S_1/S_2] = S_1 \cap V_2$ where V_2 is the Voronoi region of S_2. Codes of the form $S \cap V$ for some Voronoi region V of a sublattice of S are called *Voronoi codes.*

We see that there is a natural way to integrate the idea of Voronoi codes with our codes for the S-adder channel. Let V_i be the Voronoi region of S_i and choose the coset leaders for the partition $[S_j/S_{j+1}]$ in such a way that

$$\begin{gathered} [S_0/S_1] = S_0 \cap V_1 \\ [S_0/S_1] + [S_1/S_2] = S_0 \cap V_2 \\ \vdots \\ [S_0/S_1] + [S_1/S_2] + \cdots + [S_j/S_{j+1}] = S_0 \cap V_{j+1}. \end{gathered}$$

This implies that when a modulator uses only its input levels $0, 1, \cdots, j$, then it uses only points in the Voronoi code $S_0 \cap V_{j+1}$. The result that codes are constructed as shells of an onion. This is shown in Fig. 2 for $S = \mathsf{N}^2$, $p = 3$, and $l = 2$. In this Figure small circles denote elements of S_0, mid-sized circles elements of S_1 and large circles denote elements of S_2. Points inside the smallest square represent

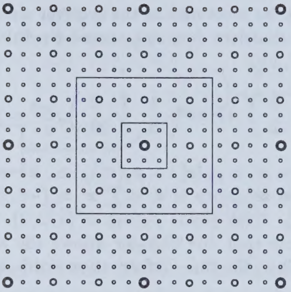


Figure 2: Example of a Voronoi code.

$[S_0/S_1] = S_0 \cap V_1$. This is the code generated by the unconstrained use of the first (from the top) input transducer's line. Mid-sized circles inside the larger square represent the coset leaders $[S_1/S_2]$. They are activated by means of the second input line. By running over all possible inputs to the first two input lines we produce the set of outputs $[S_0/S_1] + [S_1/S_2] = S_0 \cap V_2$, represented by the set of points within the larger square in Fig. 2.

The maximal rate at which information can be transmitted at level j is

$$R_j = \frac{1}{n} \log |[S_j/S_{j+1}]| = \frac{1}{n} \log p^n = \log p[\text{bits/dimension}],$$

which is independent of j. If we have T input transducers, an optimal multiple-access code operating at level j allows us to share the total rate (i.e., $(1/n)\log p$ [bits/dimension]) of level j among the T inputs. Each input can transmit an arbitrary multiple of $(1/nN)\log p$, provided that the sum does not exceed $(1/n)\log p$ [bits/dimension]. This is true provided that the power constraint of each user is large enough so that they are all allowed to use input level j. If only a subset of users has enough power to use input level j, then the same applies to that subset.

3.3 Performance

How well does the coding scheme described above compare to the ultimate limits described by the capacity region? Due to space limitations, we limit ourselves to the discussion of the case with two users. If the signal-to-noise ratio is large, i.e., $\sigma^2 << P^i$, $i = 1, 2$, and if $P^1 << P^2$, then the capacity region is closely approximated by the *pseudocapacity region* shown in Fig. 3 and defined by

$$\begin{array}{rcll} 0 & \leq & R^1 & \leq C(P^1) \\ 0 & \leq & R^2 & \leq C(P^2) \\ 0 & \leq & R^1 + R^2 & \leq C(P^2), \end{array} \tag{12}$$

where $C(P) := \frac{1}{2}\log(\frac{P}{\sigma^2})$. It is assumed that for some positive integer J_i, $i = 1, 2$, P^i equals the power P_{J_i} corresponding to the use of the modulator input lines $0, 1, \cdots, J_i$. With a sufficiently dense lattice S, our construction allows us to achieve any point in the pseudocapacity region. We see that this is substantially better than what can be achieved via time sharing (dotted line in Fig. 3). For a large population of users all having the same power constraint, our codes achieve the same rates that are achievable with time sharing, with the substantial advantage that the channel is automatically allocated to the busy users without the need for reservation or polling.

4 Conclusion

In this paper we have described a two-step technique leading to multiple-access codes for the Gaussian channel. The first step consists of constructing codes for some F-adder channel and the second step consists of mapping them into lattice codes for the Gaussian channel.

The resulting codes have the following characteristics. (1) The achievable rate region is at least as large as that of time sharing. Under certain circumstances, (see e.g. Section 3.3), the achievable rate

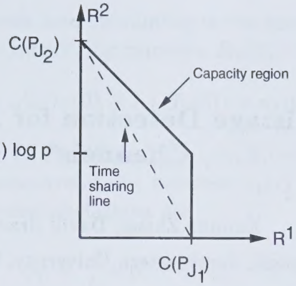


Figure 3: Pseudocapacity region.

region is very close to the capacity region. (2) The transmission rate and/or the power constraint of each user can vary over a wide range. The code of each user can be thought of as consisting of shells (layers) of increasing energy. Each layer has a maximal rate (which is the same for all layers), and each user that has enough power to use a specific layer may use it to transmit at any rate below the maximal rate. Decoding is reliable, provided that the total rate (over all users) associated to each layer is below the maximum. (3) The problems of removing the noise and separating the users are handled independently. The noise is removed in the same way as it is done for a single user, i.e., by mapping the received point to the nearest lattice point; the separation of the users is done algebraically, e.g., via the Berlekamp-Massey algorithm.

In this paper we have assumed high signal-to-noise ratios. Extensions to the low signal-to-noise case are under consideration.

References

[1] T. M. Cover and J. A. Thomas, *Elements of Information Theory.* New York: Wiley, 1991.

[2] B. S. Tsybakov, A. V. Mikhalov, and N. B. Likhanov, "Bounds for packet transmission rate in a random-multiple-access system," *Problemy Peredachi Informatsii*, vol. 19, pp. 61–81, Mar. 1983.

[3] N. Mehravari, "Random-access communication with multiple reception," *IEEE Transactions on Information Theory*, vol. 36, pp. 614–622, May 1990.

[4] R. G. Gallager, "A perspective on multiaccess channels," *IEEE Transactions on Information Theory*, vol. 31, pp. 124–142, Mar. 1985.

[5] J. H. Conway and N. J. A. Sloane, "A fast encoding method for lattice codes and quantizers," *IEEE Transactions on Information Theory*, vol. 29, pp. 820–824, November 1983.

[6] G. D. Forney, Jr. and L.-F. Wei, "Multidimensional constellations I: Introduction, figures of merit, and generalized cross constellations," *IEEE Journal of Selected Areas in Communication*, vol. 7, pp. 877–892, August 1989.

[7] G. D. Forney, Jr., "Multidimensional constellations II: Veronoi constellations," *IEEE Journal of Selected Areas in Communication*, vol. 7, pp. 941–958, August 1989.

[8] A. R. Calderbank and N. J. A. Sloane, "New trellis codes based on lattices and cosets," *IEEE Transactions on Information Theory*, vol. 33, pp. 177–195, January 1987.

Soft-Decision Multistage Detection for Asynchronous AWGN Channels*

Xuming Zhang, David Brady
ECE Department, Northeastern University, Boston, MA 02115

Abstract

Novel classes of multiuser detectors for the additive white Gaussian noise, multiple-access channel are proposed and analyzed in this paper. These classes may be described as receivers which estimate and subtract multiple-access interference (MAI) using soft, tentative decisions based on matched-filter or decorrelator outputs. Each class is specified in part by a soft-decision nonlinearity, such as a multilevel quantizer, a dead-zone nonlinearity, or a linear clipper. Each class will be fully described via the symbol error rate, as well as the exponent of this probability in the high SNR regime, or asymptotic multiuser efficiency (AME). It will be shown that the AMEs for several classes admit a simple closed form, and may be used as a design criteria to optimize the soft-decision nonlinearity. Maximal-AME detectors for each class will be compared to the matched-filter detector, the decorrelator, the hard-decision multistage detector and the maximum likelihood sequence (MLS) detector.

I Preliminaries

In a multiuser scenario, K users access a common channel and each linearly modulates a real unit-energy waveform $s_k(t), t \in [0, T)$ by a sequence of information bits $\{b_k(i) \in \{\pm 1\}, i = -M, ..., M\}$. The receiver observes the real signal

$$r(t) = \sum_{k=1}^{K} \sqrt{w_k} \sum_{i=-M}^{M} b_k(i) s_k(t - iT - \tau_k) + n(t)$$

where $n(t)$ is white Gaussian noise with power spectral density σ^2, w_k and τ_k represent the received energy and relative delay for user k. While w_k and τ_k are usually slowly time-varying, we have assumed fixed energy and relative delay over the transmission horizon for each user. Generalizations to the time-varying case are presented elsewhere [1]. We assume that accurate amplitude and relative delay estimates are possible for all users. In this case the entire sequence of matched-filter outputs forms a set of sufficient statistics to demodulate $\{b_k(i),\ \forall\, k = 1, ..., K,\ \ \forall\, i = -M, ..., M\}$. In this paper we focus on the demodulation of the 0^{th} bit of the 1^{st} user and assume $0 \leq \tau_1 \leq \cdots \leq \tau_K \leq T$ without loss. We let $\boldsymbol{B}_1(0)$ denote all symbols which interfere with $b_1(0)$ directly, i.e, $\boldsymbol{B}_1(0) = [b_2(-1), \cdots, b_K(-1), b_2(0), \cdots, b_K(0)]^T$. The normalized, matched-filter outputs for user 1 at time 0, $y_1(0)$, may be described by three terms, the last of which corresponds

---

*This work was supported in part by the Siemens Medical Corporation, and by the National Science Foundation under Grant OCE-9201191.

to the output thermal noise process, and remaining terms correspond to the influence of the desired symbol $b_1(0)$ and the interfering symbols $\boldsymbol{B}_1(0)$,

$$y_1(0) = \sqrt{w_1} b_1(0) + I_1(\boldsymbol{B}_1(0)) + n_1(0) \tag{1}$$

where the MAI term is given by $I_1(\boldsymbol{B}_1(0)) = \sum_{i=2}^{K} \sqrt{w_i}\,[b_i(-1)\rho_{i1} + b_i(0)\rho_{1i}]$ which is a linear combination of partial crosscorrelations between $s_1(t)$ and all overlapping waveforms. The normalized partial crosscorrelations are

$$\rho_{1i} = \int s_1(t-\tau_1)s_i(t-\tau_i)dt, \quad \rho_{i1} = \int s_1(t-\tau_1)s_i(t+T-\tau_i)dt, \ \ \forall\, i = 2, \cdots, K.$$

Both decision feedback and multistage multiuser detectors utilize subtractive interference rejection. In contrast to decision feedback techniques, in which decisions for interfering symbols are forced to be causally consistent, multistage detection iterates on the data decisions to be "consistent" with the previous iteration. (By "consistent" we mean that the interference is estimated using data decisions from the previous iteration). In this paper, we will focus our attention on the multistage detection strategies. The general concept is easily explained for the 2-stage detector as shown in Figure 1. The final decision statistic $z_1(0)$ and the subsequent final decision for user 1 at time 0 may be expressed as

$$z_1(0) = y_1(0) - I_1(\widetilde{\boldsymbol{B}_1}(0)) = b_1(0)w_1 + I_1(\boldsymbol{\epsilon}_1(0)) + n_1(0), \text{ and } \quad \hat{b}_1(0) = \operatorname{sgn}(z_1(0)) \tag{2}$$

where $\boldsymbol{\epsilon}_1(0)$ denotes the *tentative* decision error vector, i.e. $\boldsymbol{B}_1(0) - \widetilde{\boldsymbol{B}}_1(0)$. If the tentative decision statistics are denoted by $\{x_k(l)\}$ for all $k = 2, ..., K$; $l = -1, 0$, then the tentative decisions for the interfering symbols are $\{\tilde{b}_k(l) = g_k(x_k(l))\}$, where $g_k(x)$ is a heretofore unspecified decision function for user k. Our goal in this work is to specify $g_k(\cdot)$, $k = 2, ..., K$ which are best, in some sense, for the demodulation of user 1. In general, each user will require $K-1$ additional $g_k(\cdot)$ for demodulation. If a hard limiter is used, then we have $\epsilon_k(l) \in \{0, 2b_k(l)\}$. Note that tentative decision statistics $\{x_k(l)\}$ can be acquired by using any type of suboptimal multiuser detection strategies. Let us define the $(K-1) \times 1$ real noise vectors which are involved in the tentative decisions $\widetilde{\boldsymbol{B}}_1(0)$ as $\tilde{\boldsymbol{n}}(l), l = -1, 0$, and let $\boldsymbol{\xi}_1(0) = \left[\tilde{\boldsymbol{n}}^T(-1), n_1(0), \tilde{\boldsymbol{n}}^T(0)\right]^T$ be the entire noise vector. While this description is valid for a wide range of tentative decision statistics, we are particularly interested in 2 strategies: conventional tentative decision statistics, i.e. $x_k(l) = y_k(l)$ and L-shot decorrelated tentative decision statistics. The former set is simple to obtain, and induces a decoding delay of one symbol, while the latter trades off decoding delay ($L/2$ symbols) with improved tentative decision statistics. The L-shot decorrelator is identical to the standard decorrelator [2] for the truncated observation of L-symbols centered at $b_1(0)$ here, and is described for $L = 1$ in [3].

II Soft-tentative Decision 2-stage Detectors

In this section, we discuss the soft-decision 2-stage detection, in which the initial decisions are updated once before a final decision is reached. The generalization from 2-stage to m-stage detection is straightforward. In this paper, we will choose the tentative decision nonlinearity $g_k(\cdot)$ for the k^{th} interferer of user 1 so as to maximize the AME of user 1. Since each class of soft-decision nonlinearities will include the sign detector and the zero function, we will construct a receiver which is uniformly superior to both the hard-decision 2-stage detector and the conventional detector, with identical complexity and decoding delay as the former. For illustration, we will focus on two simple nonlinearities, i.e. a piecewise quantizer and a linear clipper as shown in Figure 2. The former is defined as (see Figure 2.(a).)

$$g_k(x) = \frac{l \cdot \mathrm{sgn}(x)}{\Gamma_k}, \quad \delta_{k,l} \le \frac{|x|}{\sqrt{w_k}} < \delta_{k,l+1}, \quad l = 0, 1, \cdots, \Gamma_k. \tag{3}$$

where $k \in \{2, 3, \cdots, K\}$ and we have set $\delta_{k,0} = 0$ and $\delta_{k,\Gamma_k+1} = \infty$ for convenience. As a result, the tentative decision vector $\widetilde{\boldsymbol{B}}_1(0)$ may take on $\prod_{k=2}^{K}(2\Gamma_k+1)^2$ possible values. Denoting the magnitude of the tentative decision $\Upsilon_k(l) = |\tilde{b}_k(l)|$, $k = 2, ..., K$, $l = -1, 0$, we define a $2(K-1) \times 1$ vector $\boldsymbol{\Upsilon} = [\Upsilon_2(-1), \cdots, \Upsilon_K(-1), \Upsilon_2(0), \cdots, \Upsilon_K(0)]^T$ and let $\Omega_{\boldsymbol{\Upsilon}}$ be the range of $\boldsymbol{\Upsilon}$, $\Omega_{\boldsymbol{\Upsilon}} = \{\boldsymbol{\Upsilon} \;:\; \Upsilon_k(\ell) \in \{0, 1/\Gamma_k, 2/\Gamma_k, \cdots, 1\};\ \forall k = 2, \cdots, K;\ \ \ell = -1, 0\}$. Let $\boldsymbol{D} = \mathrm{diag}[\boldsymbol{\Upsilon}^T]$. With this notation, the tentative decision error vector becomes $\epsilon_1(0) = [\boldsymbol{B}_1(0) - \boldsymbol{D}\boldsymbol{\beta}]$ where $\boldsymbol{\beta} \in \{\pm 1\}^{2(K-1)}$ contains the signs of the tentative decisions for the interfering symbols of $b_1(0)$. Let the $(i,j)^{th}$ entry of the $(K \times K)$ matrix $\boldsymbol{R}(l)$ be defined as $R_{i,j}(l) = \int_{-\infty}^{\infty} s_i(t-\tau_i)s_j(t-\tau_j+lT)dt$. Also, let $v_k(l)$ denote the noise term in the k^{th} user decision statistic at time l from the L-shot decorrelator output and define the $(i,j)^{th}$ component of the $K \times K$ covariance matrix $\mathcal{D}(l)$ as $\mathcal{D}_{i,j}(l) = E[v_i(0)v_j^*(l)]$. Then the covariance matrices of the zero mean Gaussian noise vectors $\boldsymbol{\xi}_1(0)$ for the conventional and L-shot decorrelated tentative decision statistics are, respectively,

$$\Lambda = \begin{pmatrix} \widetilde{\boldsymbol{R}}(0) & \widetilde{\boldsymbol{R}}(-1) \\ \widetilde{\boldsymbol{R}}(1) & \boldsymbol{R}(0) \end{pmatrix} \sigma^2, \quad \Lambda^d = \begin{pmatrix} \widetilde{\mathcal{D}}(0) & 0 & \widetilde{\mathcal{D}}(-1) \\ 0^T & 1 & 0^T \\ \widetilde{\mathcal{D}}(1) & 0 & \widetilde{\mathcal{D}}(0) \end{pmatrix} \sigma^2$$

where $\widetilde{\boldsymbol{R}}(0)$ is a $(K-1) \times (K-1)$ matrix obtained by deleting the first column and the first row of $\boldsymbol{R}(0)$, $\widetilde{\boldsymbol{R}}(1)$ is defined by deleting the first column of $\boldsymbol{R}(1)$, $\widetilde{\boldsymbol{R}}(1) = \widetilde{\boldsymbol{R}}(-1)^T$, and $(K-1) \times (K-1)$ matrices $\widetilde{\mathcal{D}}(l), \forall l = 0, \pm 1$ are derived from $\mathcal{D}(l)$ by deleting the first row and first column.

The second proposed nonlinearity for tentative decisions is a linear clipper (see Figure 2.(b))

$$g_k(x) = \begin{cases} \mathrm{sgn}(x) & \frac{|x|}{\sqrt{w_k}} \ge \delta_k \\ \frac{x}{\delta_k \sqrt{w_k}} & \frac{|x|}{\sqrt{w_k}} < \delta_k \end{cases} \tag{4}$$

We will restrict our attention to the two-user case for this class of detectors and consider the design of $g_2(\cdot)$ for interference suppression of user 1. We shall also consider conventional tentative decision statistics for the remainder of this section. According to $g_2(x)$ in (4), the tentative decision regions for $(\tilde{b}_2(-1),\ \tilde{b}_2(0))$ are divided into four parts as

$$\Big(\tilde{b}_2(-1), \tilde{b}_2(0)\Big) = \begin{cases} (i, j), & \text{if } \ (y_2(-1), y_2(0)) \in R_{dd}, \\ (\frac{y_2(-1)}{\delta_2\sqrt{w_2}}, j), & \text{if } \ (y_2(-1), y_2(0)) \in R_{cd}, \\ (i, \frac{y_2(0)}{\delta_2\sqrt{w_2}}), & \text{if } \ (y_2(-1), y_2(0)) \in R_{dc}, \\ (\frac{y_2(-1)}{\delta_2\sqrt{w_2}}, \frac{y_2(0)}{\delta_2\sqrt{w_2}}), & \text{if } \ (y_2(-1), y_2(0)) \in R_{cc}, \end{cases}$$

where $i, j \in \{\pm 1\}$. It is easily shown that the final decision statistics for these four tentative decision regions can be expressed as, respectively,

$$\begin{aligned} z_1(0) = b_1(0)\sqrt{w_1} + (b_2(-1) - i)\sqrt{w_2}\rho_{21} + (b_2(0) - j)\sqrt{w_2}\rho_{12} \\ + n_{dd}(0), \ \text{ if } \ (y_2(-1), y_2(0)) \in R_{dd}, \end{aligned} \tag{5}$$

$$\begin{aligned} z_1(0) &= b_1(0)\left[1-\gamma_2\rho_{21}^2\right]\sqrt{w_1}+b_2(-1)[1-\gamma_2]\sqrt{w_2}\rho_{21}+(b_2(0)-j)\sqrt{w_2}\rho_{12} \\ &- b_1(-1)\sqrt{w_1}\rho_{12}\rho_{21}\gamma_2+n_{cd}(0), \text{ if } (y_2(-1),y_2(0))\in R_{cd}, \end{aligned} \tag{6}$$

$$\begin{aligned} z_1(0) &= b_1(0)\left[1-\gamma_2\rho_{12}^2\right]\sqrt{w_1}+(b_2(-1)-i)\sqrt{w_2}\rho_{21}+b_2(0)[1-\gamma_2]\sqrt{w_2}\rho_{12} \\ &- b_1(0)\sqrt{w_1}\rho_{12}\rho_{21}\gamma_2+n_{dc}(0), \text{ if } (y_2(-1),y_2(0))\in R_{dc}, \end{aligned} \tag{7}$$

and

$$\begin{aligned} z_1(0) &= b_1(0)\left[1-\gamma_2(\rho_{12}^2+\rho_{21}^2)\right]\sqrt{w_1}+b_2(-1)[1-\gamma_2]\sqrt{w_2}\rho_{21}+b_2(0)[1-\gamma_2]\sqrt{w_2}\rho_{12} \\ &- [b_1(-1)+b_1(1)]\sqrt{w_1}\rho_{12}\rho_{21}\gamma_2+n_{cc}(0), \text{ if } (y_2(-1),y_2(0))\in R_{cc}, \end{aligned} \tag{8}$$

where $\gamma_2=\frac{1}{\delta_2}$ and the new noise terms

$$\begin{aligned} n_{dd}(0)=n_1(0), \qquad\qquad & n_{cd}(0)=n_1(0)-n_2(-1)\rho_{21}\gamma_2, \\ n_{dc}(0)=n_1(0)-n_2(0)\rho_{12}\gamma_2, \quad & n_{cc}(0)=n_1(0)-(n_2(-1)\rho_{21}+n_2(0)\rho_{12})\gamma_2 \end{aligned}$$

are Gaussian with zero mean and with variances respectively

$$\begin{aligned} \mathcal{K}_{dd}\sigma^2=\sigma^2, \qquad\qquad & \mathcal{K}_{cd}\sigma^2=(1+\rho_{21}^2\gamma_2^2-2\rho_{21}^2\gamma_2)\sigma^2, \\ \mathcal{K}_{dc}\sigma^2=(1+\rho_{12}^2\gamma_2^2-2\rho_{12}^2\gamma_2)\sigma^2, \quad & \mathcal{K}_{cc}\sigma^2=(1+(\rho_{21}^2+\rho_{12})(\gamma_2^2-2\gamma_2))\sigma^2. \end{aligned}$$

Accordingly, the noise vectors for each of the four cases, i.e. $\boldsymbol{\xi}_{dd}(0)$, $\boldsymbol{\xi}_{cd}(0)$, $\boldsymbol{\xi}_{dc}(0)$ and $\boldsymbol{\xi}_{cc}(0)$ are Gaussian with zero mean vectors and covariance matrices

$$\Lambda_{dd}=\begin{pmatrix} 1 & \rho_{21} & 0 \\ \rho_{21} & \mathcal{K}_{dd} & \rho_{12} \\ 0 & \rho_{12} & 1 \end{pmatrix}\sigma^2, \quad \Lambda_{cd}=\begin{pmatrix} 1 & \rho_{21}(1-\gamma_2) & 0 \\ \rho_{21}(1-\gamma_2) & \mathcal{K}_{cd} & \rho_{12} \\ 0 & \rho_{12} & 1 \end{pmatrix}\sigma^2,$$

$$\Lambda_{dc}=\begin{pmatrix} 1 & \rho_{21} & 0 \\ \rho_{21} & \mathcal{K}_{dc} & \rho_{12}(1-\gamma_2) \\ 0 & \rho_{12}(1-\gamma_2) & 1 \end{pmatrix}\sigma^2,$$

and

$$\Lambda_{cc}=\begin{pmatrix} 1 & \rho_{21}(1-\gamma_2) & 0 \\ \rho_{21}(1-\gamma_2) & \mathcal{K}_{cc} & \rho_{12}(1-\gamma_2) \\ 0 & \rho_{12}(1-\gamma_2) & 1 \end{pmatrix}\sigma^2.$$

The case of decorrelated tentative decision statistics is quite similar and is not included here. In next section, we will derive the probability of error and asymptotic multiuser efficiency for these soft-decision scenarios.

III Symbol Error Probability and AME for Soft-decision 2-stage Detectors

The AME for hard-decision multistage detectors was derived in [4] and [5]. Following the derivation similar to [5], we define $\boldsymbol{\mathcal{B}}_1(0)$ as all symbols involved in the tentative decision statistics and not in the final decision statistics for the demodulation of $b_1(0)$, i.e. $\boldsymbol{\mathcal{B}}_1(0)=[b_2(-2),\cdots,b_K(-2),b_1(-1),b_1(1),\cdots,b_{K-1}(1)]^T$. Define $\boldsymbol{B}_k(\ell)$ and $I_k(\boldsymbol{B}_k(\ell))$ for all $k=2,\dots,K$, $\ell=-1,0$, similar to $\boldsymbol{B}_1(0)$ and $I_1(\boldsymbol{B}_1(0))$ respectively. The residual MAI for a quantizer nonlinearity is given by

$$I_1(\epsilon_1(0))=\sum_{i=2}^{K}\left[b_i(-1)-\beta_{i-1}\Upsilon_i(-1)\right]\sqrt{w_i}\rho_{i1}+\sum_{j=2}^{K}\left[b_j(0)-\beta_{K+j-1}\Upsilon_j(0)\right]\sqrt{w_j}\rho_{1j} \tag{9}$$

where $\Upsilon_i(-1) = \frac{m_i}{\Gamma_i}, \Upsilon_j(0) = \frac{l_j}{\Gamma_j}$, $m_i,\ l_i \in \{0, \cdots, \Gamma_i\}$; $\forall i = 2, ..., K$ and $\boldsymbol{\beta}$ is. as before, the signs of the tentative decisions. If we define $\alpha_k(l) = -\sqrt{w_k} b_k(l) - I_k(\boldsymbol{B}_k(l))$, then the probability of the event $\{b_1(0) = +1\}$ given $\{b_1(0) = -1\}$ may be given as

$$\Pr[error] = E_{\mathcal{B}_1(0), \boldsymbol{B}_1(0)} E_{\boldsymbol{\xi}_1(0)} \sum_{\boldsymbol{\beta} \in \{\pm 1\}^{2(K-1)}, \boldsymbol{\Upsilon} \in \Omega_{\boldsymbol{\Upsilon}}} \Big[\mathbb{1}\{n_1(0) > \sqrt{w_1} - I_1(\epsilon_1(0))\} \times$$

$$\prod_{i=2}^{K} \mathbb{1}\{\delta_{i,m_i}\sqrt{w_i} + \beta_{i-1}\alpha_i(-1) < \beta_{i-1} n_i(-1) < \delta_{i,m_i+1}\sqrt{w_i} + \beta_{i-1}\alpha_i(-1)\}$$

$$\prod_{j=2}^{K} \mathbb{1}\{\delta_{j,l_j}\sqrt{w_j} + \beta_{K-2+j}\alpha_j(0) < \beta_{K-2+j} n_j(0) < \delta_{j,l_j+1}\sqrt{w_j} + \beta_{K-2+j}\alpha_j(0)\} \Big] \quad (10)$$

where $\mathbb{1}\{A\}$ is the indicator function of the event A, $E_{\mathcal{B}_1(0), \boldsymbol{B}_1(0)}$ denotes the expectation over the indexing variates and $E_{\boldsymbol{\xi}_1(0)}$ denotes the expectation over the joint distribution of the noise vector $\boldsymbol{\xi}_1(0)$, which is independent of $\boldsymbol{B}_1(0)$ and $\mathcal{B}_1(0)$. It can be shown that similar results may be obtained for tentative decisions based on decorrelator outputs. Obviously, in both cases the numerical evaluation of the exact error rate involves the $(2K - 1)$-dimensional normal distribution function over a number of terms which is exponential in the number of users (the cardinality of $\Omega_{\boldsymbol{\Upsilon}}$ is $\prod_{i=2}^{K}(\Gamma_k + 1)^2$).

Now, we briefly consider tentative decisions based on a linear clipper. Again, we only consider the two-user case to simplify the exposition. Since the tentative decision regions R_{dd}, R_{cd}, R_{dc}, R_{cc} are disjoint and exhaustive, the error probability for $b_1(0)$ may be expressed in four terms

$$\Pr[\widehat{b_1(0)} \neq b_1(0)] = \Pr[error, (y_2(-1), y_2(0)) \in R_{dd}] + \Pr[error, (y_2(-1), y_2(0)) \in R_{cd}]$$
$$+ \Pr[error, (y_2(-1), y_2(0)) \in R_{dc}] + \Pr[error, (y_2(-1), y_2(0)) \in R_{cc}]. \quad (11)$$

Since the final decision statistics, tentative decision statistics, and all relevant noise vectors are obtained in the Section III, the error probability expression for all terms in (11) follows similarly as in (10).

In order to derive the AME for either class of soft-tentative-decision, 2-stage detectors, we find it more convenient to relate the expectations over the noise vector $\boldsymbol{\xi}_1(0)$ to Gaussian vectors with independent, identically distributed components. Consider a square-root factorization of the covariance matrix of $\boldsymbol{\xi}_1(0)$, $\sigma^2 \mathbf{S}\mathbf{S}^T$, and let the i^{th} row of $\mathbf{S}$ be denoted by $\mathbf{S}_i^T$. Then there exists a Gaussian vector $\mathbf{q}$, with zero mean vector and covariance matrix $\sigma^2 \boldsymbol{I}$ such that $\boldsymbol{\xi}_1(0)$ has the same distribution as $\mathbf{Sq}$ and thus the error event for the general term in the error rate expression (10) (indexed by $\boldsymbol{B}_1(0)$, $\boldsymbol{\beta}$, and $\boldsymbol{\Upsilon}$) is

$$\Psi(\boldsymbol{B}_1(0), \boldsymbol{\beta}, \boldsymbol{\Upsilon}) \triangleq \{\mathbf{S}_K^T \mathbf{q} \geq \sqrt{w_1} - I_1(\epsilon_1(0))\} \cap$$
$$\bigcap_{i=2}^{K} \left\{\delta_{i,m_i}\sqrt{w_i} + \beta_{i-1}\alpha_i(-1) \leq \beta_{i-1}\mathbf{S}_{i-1}^T \mathbf{q} < \delta_{i,m_i+1}\sqrt{w_i} + \beta_{i-1}\alpha_i(-1)\right\}$$
$$\bigcap_{j=2}^{K} \left\{\delta_{j,l_j}\sqrt{w_j} + \beta_{K+j-1}\alpha_j(0) \leq \beta_{K+j-1}\mathbf{S}_{K+j-1}^T \mathbf{q} < \delta_{j,l_j+1}\sqrt{w_j} + \beta_{K+j-1}\alpha_j(0)\right\}$$

where $\boldsymbol{\Upsilon} \in \Omega_{\boldsymbol{\Upsilon}}$, $\boldsymbol{\beta} \in \{\pm 1\}^{2K-1}$ and $\delta_{l,j}$ are constants. Similar expressions are obtained for the 2-stage detector with tentative decisions via a linear clipper (i.e., for the 2-user case we may express the error event in similar way for each of the four different noise vectors). The following lemma is crucial to the solution of the asymptotic multiuser efficiency[4].

Lemma 1: Let Ξ denote the support of $\mathbb{1}\{\Psi(\boldsymbol{B}_1(0), \boldsymbol{\beta}, \boldsymbol{\Upsilon})\}$ in the space $(\mathrm{q}_1, \cdots, \mathrm{q}_{2K-1})$. Then,

$$\lim_{\sigma \to 0} \frac{-\log \Pr[\mathrm{q} \in \Xi]}{\frac{w_1}{2\sigma^2}} = \vartheta(\boldsymbol{B}_1(0), \boldsymbol{\beta}, \boldsymbol{\Upsilon}), \text{ where } \vartheta(\boldsymbol{B}_1(0), \boldsymbol{\beta}, \boldsymbol{\Upsilon}) = \min_{\mathrm{q} \in \Xi} \frac{\|\mathrm{q}\|^2}{w_1},$$

$$\eta_1 = \inf_{\substack{\boldsymbol{B}_{1(0)}, \boldsymbol{\beta} \in \{\pm 1\}^{4(K-1)} \\ \boldsymbol{\Upsilon} \in \Omega_{\boldsymbol{\Upsilon}}}} \vartheta(\boldsymbol{B}_1(0), \boldsymbol{\beta}, \boldsymbol{\Upsilon}).$$

□

This lemma relates the error rate exponent of each term in the error probability expression to a minimum norm problem with linear inequality constraints. If the worst-case constrained region contains the origin, then the AME is zero. Otherwise, the AME is positive. In what follows we define $\mu = \sqrt{\frac{w_2}{w_1}}$, $p_{ij} = |\rho_{ij}|$, $p_S = \min\{p_{12}, p_{21}\}$ and $p_L = \max\{p_{12}, p_{21}\}$ for convenience.

Special Cases - 2 Asynchronous Users

(a). Soft-decision 2-stage Detection Using a Dead-zone Limiter

Of special interest, we particularize the general multilevel quantizer to the simplest nontrivial example – a dead-zone limiter ($\Gamma_k = 1$ and in the sequel, we use δ_2 instead of $\delta_{2,1}$ for convenience). The weakness of employing hard tentative decisions ($\delta_2 = 0$) in the traditional 2-stage detector is emphasized when the tentative decision statistic is very small in magnitude, since a decision based on the sign of $x_2(l)$ is most likely to be incorrect, due to the effects of random noise and MAI. By comparison, the dead-zone device simply mutes the signal instead of passing the wrong tentative decision to the final stage. The tentative decision error $\epsilon_k(l)$ takes on $\{2b_k(l), b_k(l), 0\}$ corresponding to incorrect, suppressed or correct tentative decisions, respectively. The closed form expressions of AME for hard-decision 2-stage detection have been derived in [5]. It is possible but tedious to obtain closed-form expressions for the AME for 2-stage detection with a dead-zone limiter based on conventional tentative decision statistics. However, for the dead-zone limiter with L-shot decorrelated tentative decision statistics, the AME may be expressed in closed-form as:

lemma 2: Let $\eta_{2,L}$ denote the AME for user 2 in the L-shot sliding-window decorrelator, and define

$$\begin{aligned}
\vartheta_a &= \max^2\{0, 1 - 2\mu p_{12} - 2\mu p_{21}\} + \frac{2(1+\delta_2)^2 \eta_{2,L} \mu^2}{1+\zeta}, \\
\vartheta_b &= \max^2\{0, 1 - 2\mu p_L - \mu p_S\} + \begin{cases} \frac{2[1+\delta_2^2-(1-\delta_2^2)\zeta]\eta_{2,L}\mu^2}{1-\zeta^2}, & 0 \le \delta_2 < \frac{1-\zeta}{1+\zeta} \\ (1+\delta_2)^2 \eta_{2,L} \mu^2, & \delta_2 \ge \frac{1-\zeta}{1+\zeta}; \end{cases} \\
\vartheta_c &= \max^2\{0, 1 - 2\mu p_L\} + (1+\delta_2)^2 \eta_{2,L} \mu^2, \\
\vartheta_d &= \max^2\{0, 1 - \mu p_{12} - \mu p_{21}\} + \frac{2\max^2\{0, 1-\delta_2\}}{1+\zeta} \eta_{2,L} \mu^2; \\
\vartheta_e &= \max^2\{0, 1 - \mu p_L\} + \max^2\{0, 1-\delta_2\} \eta_{2,L} \mu^2.
\end{aligned}$$

where $\eta_{2,L} = \frac{1}{\mathcal{D}_{2,2}(0)}$ and $\zeta = \frac{\mathcal{D}_{2,2}(1)}{\mathcal{D}_{2,2}(0)}$. Then, the asymptotic multiuser efficiency for soft-decision 2-stage detection with a dead-zone limiter based on L-shot decorrelated tentative decision statistics is $\eta_1(\delta_2) = \min\{\vartheta_a, \vartheta_b, \vartheta_c, \vartheta_d, \vartheta_e, 1\}$. □

Proof: See [5] –A similar proof can be found therein for hard-decision 2-stage detection with decorrelated tentative decision statistics. It can be verified that for this type detector, the minimum value of $\eta_1(\delta_2)$ is less than $\eta_{2,L}$ due to the tradeoff characteristic of a dead-zone type device. □

(b). Soft-decision 2-stage Detection Using a Linear Clipper

We have characterized the worst-case performance (over $\mu = \sqrt{\frac{w_2}{w_1}}$) of the optimal detector of this class.

Lemma 3: Consider the demodulation of user 1 in an asynchronous, 2-user channel. For soft-decision, 2-stage detection employing conventional tentative decision statistics and a linear clipper, we have

$$\bar{\eta}_{opt} \overset{\Delta}{=} \inf_{\mu \geq 0} \sup_{\delta_2 \geq 0} \eta_1(\delta_2) = \frac{(1-(p_{12}+p_{21})^2)^2}{1-p_{12}^2-p_{21}^2}; \tag{12}$$

and for the L-shot decorrelated tentative decision statistics using a linear clipper, we have

$$\inf_{\mu \geq 0} \sup_{\delta_2 \geq 0} \eta_1(\delta_2) = \eta_{2,L}. \tag{13}$$

□

The proof of Lemma 3 is quite involved and is omitted here due to the limitations of space. Intuitively, the lemma characterizes the near-far resistance of these detectors for slowly time-varying channels. It is also of interest to consider tentative decisions using an adaptive amplifier (without clipping the signals) with gain equal to $\frac{1}{\sqrt{w_2}}$. It is easy to check that the AMEs for user 1 are exactly the same as (12) and (13) for the conventional and L-shot decorrelated tentative decision statistics, respectively.

(c). Optimal Strategies v.s. Suboptimal Approximations to δ_2

As described above, the soft-decision 2-stage detection using a dead-zone limiter or a linear clipper encompasses: (i). conventional matched-filter detection with $\delta_2 = \infty$; (ii). hard-decision 2-stage detection [6] with $\delta_2 = 0$. More importantly, there exists an optimal strategy (δ_2) among each class, which also depends on the preprocessing (conventional or decorrelated tentative decision statistics). This optimal strategy is specified by which δ_2 maximizes the AME for the desired user (say user 1) such that

$$\delta_{2,opt} = \arg\max_{\delta_2 \geq 0} \eta_1(\delta_2).$$

Explicit expressions of $\delta_{2,opt}$ are very difficult to obtain, but may be approximated closely using Lemma 2. In the following, we present suboptimal δ_2's with explicit expressions that achieve satisfactory AMEs. Empirically, we have used $\delta_2 = \left(\frac{(p_{12}+p_{21})^\alpha}{\mu}\right)^\beta$ and $\delta_2 = \left(\frac{p_L^\alpha}{\mu}\right)^\beta$ to approximate $\delta_{2,opt}$ for the conventional and L-shot decorrelated tentative decision statistics respectively (α's and β's are given in Table 1 for each class of soft-decision 2-stage detector). The performance of detectors with these suboptimal δ_2's will be evaluated in the next section.

Table 1: Empirical α's and β's for suboptimal δ_2's for a dead-zone limiter.

Conventional Tentative Decisions		Decorrelated Tentative Decisions	
$(p_{12}+p_{21})^2 \leq \frac{1}{2}$	$(p_{12}+p_{21})^2 > \frac{1}{2}$	$2p_L \leq 1$	$2p_L > 1$
$\alpha = 1.0 \sim 2.2$	$\alpha = 1.5 \sim 2.5$	$\alpha = 0.9 \sim 1.2$	$\alpha = 1.2 \sim 1.8$
$\beta = 1.5 \sim 3.5$	$\beta = 3.5 \sim 4.5$	$\beta = 2 \sim 4$	$\beta = 3.5 \sim 5.0$

Suboptimal expressions for δ_2 for soft-decision 2-stage detection with a linear clipper are $\delta_2 = \infty$ for small μ, 0 for large μ and otherwise elect $\delta_2 = 1$. In this case, it is easy to show that the AME η_1 exceeds that of conventional or hard-decision 2-stage detection. The suboptimal δ_2 can be derived for the linear-clipper 2-stage detector with conventional tentative decision statistics (where $\bar{\eta}_{1,opt}$ is defined in (12))

$$\text{Case 1: } (p_{12}+p_{21}) > \tfrac{1}{2p_L}, \qquad \text{Case 2: } \tfrac{1}{2p_{12}+2p_{21}} < p_{12}+p_{21} \le \tfrac{1}{2p_L}$$

$$\delta_2 = \begin{cases} \infty, & 0 \le \mu < \frac{1-\sqrt{\bar{\eta}_{2,opt}}}{p_{12}+p_{21}}, \\ 0, & \mu \ge u_1, \\ 1, & \text{otherwise.} \end{cases}, \quad \delta_2 = \begin{cases} \infty, & 0 \le \mu < \frac{1-\sqrt{\bar{\eta}_{2,opt}}}{p_{12}+p_{21}}, \\ 0, & \mu \ge u_2, \\ 1, & \text{otherwise.} \end{cases} \tag{14}$$

where $u_1 = \sqrt{\bar{\eta}_{1,opt}} + p_{12} + p_{21}$, and u_2 is determined by

$$u_3 = \begin{cases} u_1, & \text{if } u_1 >= \frac{1}{p_L} - (p_{12}+p_{21}), \\ -b + \sqrt{b^2 - c}, & \text{otherwise} \end{cases}$$

where $b = 2p_L^2(p_{12}+p_{21}) - (p_{12}+p_{21}) - p_L$ and $c = 1 - 2p_L(p_{12}+p_{21}) + (p_{12}+p_{21})^2 - (1-p_L^2)\bar{\eta}_{opt}$. Similarly, for the case of the linear-clipper 2-stage detector with decorrelated tentative decision statistics, a suboptimal expression for δ_2 is given by: (where $\eta_{2,L}$ is the AME for user 2 in the L-shot decorrelation)

$$\text{Case 1: } 2p_L \ge 1, \qquad \text{Case 2: } 2p_L 2 < 1$$

$$\delta_2 = \begin{cases} \infty, & 0 \le \mu < \frac{1-\sqrt{\eta_{2,L}}}{p_{12}+p_{21}}, \\ 0, & \mu \ge 1, \\ 1, & \text{otherwise.} \end{cases} . \quad \delta_2 = \begin{cases} \infty, & 0 \le \mu < \frac{1-\sqrt{\eta_{2,L}}}{p_{12}+p_{21}}, \\ 0, & \mu \ge \frac{2p_L + \sqrt{4\eta_{2,L}p_L^2 + \eta_{2,L}^2 - \eta_{2,L}}}{\eta_{2,L}+4p_L^2}, \\ 1, & \text{otherwise.} \end{cases} \tag{15}$$

(d). M-stage Detectors with $M > 2$

The proposed soft-decision schemes and their performance analysis can be easily generalized to m-stage detection with $m > 2$. It can be verified by induction that the following result is true for any m-stage detector with $m > 2$ [5].

Lemma 4: The AME of a fixed user for m-stage detection is uniformly larger than that for $(m-2)$-stage detection using the same tentative decision technique. □

IV Numerical Results

Figure 3 shows, for 2 asynchronous users, the AMEs versus the relative energy of the interferer for 2-stage detectors using a dead-zone limiter (suboptimal δ's are computed using $\alpha = 2$, $\beta = 4$ for conventional and $\alpha = 1.5$, $\beta = 4$ for decorrelated tentative decision statistics). Figure 4 illustrates the AMEs for 2-stage detectors using a linear clipper. Also shown in both figures are the AMEs for the optimal sequence detector (MLS) [7], as well as the AME of the ∞-shot decorrelator, conventional detector and the comparable hard-decision 2-stage detector. Figure 5 shows the AMEs of soft-decision 2-stage detection using a dead-zone limiter with suboptimal δ_2 ($\alpha = 1.5$, $\beta = 4$) via L-shot decorrelated tentative decision statistics for $L = 1, 2, 3$, and ∞ where $E_1 = \int_{\tau_2}^{T-\tau_1} s_1^2(t)dt$. It is shown in Figure 5 that a decoding delay of 2 symbols (L=3) yields an AME sufficiently close to that of ∞-shot decorrelated tentative decision statistics. The next example is aimed at demonstrating the AMEs for soft-decision m-stage detectors for $m = 2, 3$ as shown in Figure 6, where for the 3-stage detectors we use suboptimal 2-stage soft-decisions as tentative decisions (Note that $\delta_1 = ((p_{12}+p_{21})^\alpha \mu)^\beta$ in this case, where α's and β's are the same as those above for these two tentative decision statistics.). Note that m-stage detection with *m even* can do better for weak users while *m odd* can perform better only for strong users. The last example (Figure 7) shows the symbol error probabilities for 2-stage detections based on conventional tentative decision statistics using a hard-limiter, a dead-zone limiter and a linear clipper with optimal δ_2's, and for conventional detection and the single-user bound. Note that the error rate floor exhibited by the hard-decision 2-stage detector is removed by the introduction of soft-tentative decisions, without incurring an increase in decoding delay or demodulation complexity.

References

[1] D. Brady and J. Catipovic, "An Adaptive, Soft-Decision Multiuser Receiver for Underwater Acoustical Channels," in *Proceedings of the Asilomar Conference on Signals Systems and Computers,* Pacific Grove, pp.1137-1141. Nov. 1992

[2] R. Lupas and S. Verdu, "Near-Far Resistance of Multiuser Detectors in Asynchronous Channels'," *IEEE Trans. Communications,* vol. Com-38, pp.496-508, April 1990.

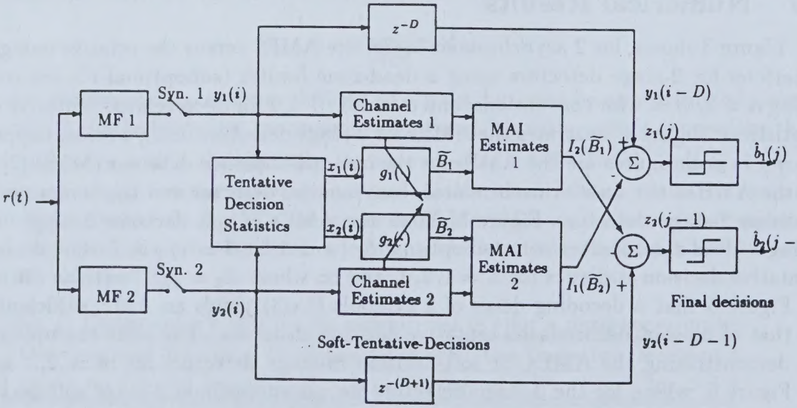
[3] S. Verdu, " Recent Progress in Multiuser Detection,", *Advance in Communications and Control Systems, ComCon 88',* Lecture Notes in Control and Information Sciences Series, NY: Springer-Verlag, 1989.

[4] D. Brady, "Asymptotic Multiuser Efficiency for Two-stage Detectors in AWGN Channels," *Proceedings of 1993 IEEE International Symposium on Information Theory.*

[5] X. Zhang and D. Brady "Asymptotic Multiuser Efficiency for M-stage Detection", submitted to *IEEE Trans. Information Theory,* 1993

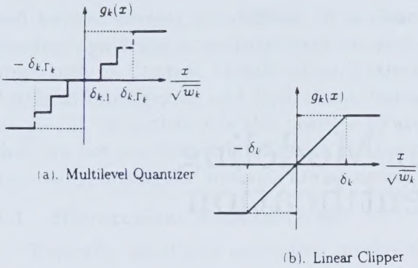
[6] M. Varanasi and B. Aazhang, "Multistage Detection in Asynchronous Code-Division Multiple-Access Channels," *IEEE Trans. Communications,* vol. COM-38, pp.509-519, April 1990.

[7] S. Verdu, "Optimum Multiuser Asymptotic Efficiency," *IEEE Trans. Communications,* vol. Com-34, pp.890-897. Sept. 1986.



z^{-D}
Syn. 1 $y_1(i)$
MF 1
Channel Estimates 1
MAI Estimates 1
$y_1(i-D)$
$z_1(j)$
$I_2(\widetilde{B}_1)$
+
-
Σ
$\hat{b}_1(j)$
$x_1(i)$
$g_1(\cdot)$
$\widetilde{B}_1$
$r(t)$
Tentative Decision Statistics
$x_2(i)$
$g_2(\cdot)$
$\widetilde{B}_2$
$z_2(j-1)$
$\hat{b}_2(j-1)$
MAI Estimates 2
Syn. 2
MF 2
Channel Estimates 2
$y_2(i)$
$I_1(\widetilde{B}_2)$ +
Final decisions
Soft-Tentative-Decisions
$y_2(i-D-1)$
$z^{-(D+1)}$

Figure 1: Block Diagram of Soft-Decision 2-stage Detector



(a). Multilevel Quantizer

(b). Linear Clipper

Figure 2: Proposed Soft-Decision Nonlinearities

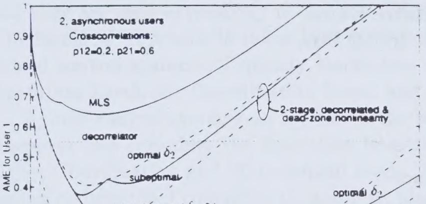


Figure 3: AME for 2-asynchronous users versus relative energy of interferer for soft-decision 2-stage detectors using a dead-zone limiter with optimal δ_2s (optimal) and suboptimal δ_2s (suboptimal). The MLS detector, decorrelator, conventional and hard-decision 2-stage detectors are shown for reference.

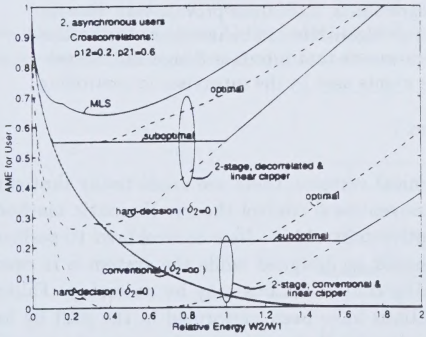


Figure 4: AME for 2-asynchronous users versus relative energy of interferer for soft-decision 2-stage detectors using a linear clipper with optimal δ_2s (optimal) and suboptimal δ_2s (suboptimal). The MLS detector, conventional and hard-decision 2-stage detectors are shown for reference.

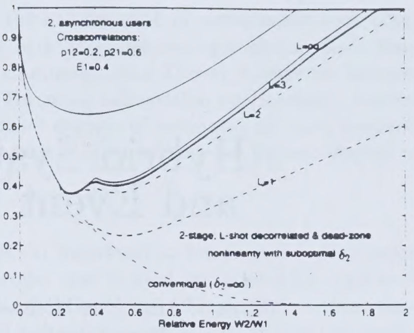


Figure 5: AME for soft-decision 2-stage detectors with the same dead-zone limiter (Table 1 $\alpha = 1.5, \beta = 4$) via L-shot decorrelated tentative decisions for $L = 1, 2, 3$ and ∞.

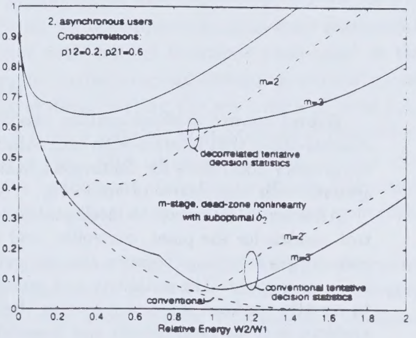


Figure 6: AME for soft-decision m-stage detectors for $m = 2, 3$ using a suboptimal dead-zone limiter via conventional and infinite-horizon decorrelated tentative decisions

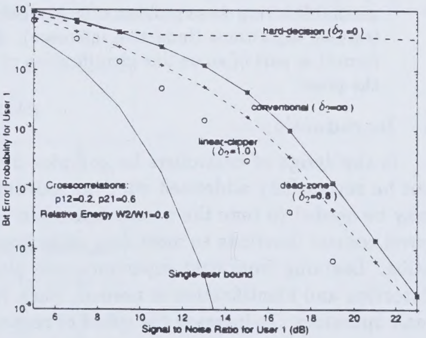


Figure 7: Symbol error probability for 2-stage detectors using a hard-limiter, an optimal dead-zone limiter, an optimal linear clipper via conventional tentative decision statistics, and for conventional and optimal isolated single-user detectors

Hybrid System Modeling and Event Identification

Panos J. Antsaklis, Michael Lemmon, and James A. Stiver
Department of Electrical Engineering
University of Notre Dame
Notre Dame, IN 46556 USA

Abstract

Hybrid control systems contain two distinct types of systems, continuous state and discrete-state, that interact with each other. Their study is essential in designing sequential supervisory controllers for continuous- state systems, and it is central in designing control systems with high degree of autonomy.

After an introduction to intelligent autonomous control and its relation to hybrid control, models for the plant, controller, and interface are introduced. The interface contains memoryless mappings between the supervisor's symbolic domain and the plant's nonsymbolic state space. The simplicity and generality afforded by the assumed interface allows us to directly confront important system theoretic issues in the design of supervisory control systems; such as determinism and quasideterminism. It is possible to formulate sufficient conditions which guarantee that the equivalent plant DES satisfies specified deterministic or quasi-deterministic relationships. These conditions are linear inequalities which can be solved for using well known convex programming algorithms such as the ellipsoid method or interior point algorithms. These convex programming algorithms provide very efficient means of solving these problems. In addition, these algorithms can be interpreted as on-line learning algorithms (inductive inference), which means that interface design can be performed as part of an on-line identification of the events used by the supervisor in controlling the plant.

1 Introduction

In the design of controllers for complex dynamical systems, there are needs today that cannot be successfully addressed with the existing conventional control theory. Heuristic methods may be needed to tune the parameters of an adaptive control law. New control laws to perform novel control functions to meet new objectives should be designed while the system is in operation. Learning from past experience and planning control actions may be necessary. Failure detection and identification is needed. Such functions have been performed in the past by human operators. To increase the speed of response, to relieve the operators from mundane tasks, to protect them from hazards, a high degree of autonomy is desired. To achieve this autonomy, high level decision making techniques for reasoning under uncertainty must be utilized. These techniques, if used by humans, may be attributed to intelligence. Hence, one way to achieve high degree of autonomy is to utilize high level decision making techniques, intelligent methods, in the autonomous controller. In our view, higher autonomy is the objective, and intelligent controllers are one way to achieve it. The need for quantitative methods to model and analyze the dynamical behavior of such autonomous systems presents significant challenges well beyond current capabilities. It is clear that the development of autonomous controllers requires significant interdisciplinary research effort as it integrates concepts and methods from areas such as Control, Identification, Estimation, Communication Theory, Computer Science, Artificial Intelligence, and Operations Research. For more information on intelligent control see [1–12]. Note that it is also possible to attain higher degrees of autonomy by using methods that are not considered intelligent. It appears however that to achieve the highest degrees of autonomy, intelligent methods are necessary indeed.

1.1 Hierarchical Architectures

Typically, intelligent controllers involve hierarchical functional architectures. A description of a hierarchical functional architecture of a controller that is used to attain high degrees of autonomy in future space vehicles can be found in [2]. Hierarchies offer very convenient ways to describe the operation of complex systems and deal with computational complexity issues, and they are used extensively in the modeling of intelligent autonomous control systems. Such a hierarchical approach is taken here (and in [2] and [4]) to study intelligent autonomous and hybrid control systems. Typically hierarchies for intelligent systems consist of three levels, the Execution Level, the Coordination Level, and the Management and Organization Level [8–11].

Hybrid control systems do appear in the intelligent autonomous control system framework whenever one considers the Execution level together with control functions performed in the higher Coordination and Management levels. Examples include expert systems supervising and tuning conventional controller parameters, planning systems setting the set points of local control regulators, sequential controllers deciding which from a number of conventional controllers is to be used to control a system, to mention but a few. One obtains a hybrid control system of interest whenever one considers controlling a continuous-state plant (in the Execution level) by a control algorithm that manipulates symbols, that is by a discrete-state controller (in Coordination and/or Management levels).

In the remainder of this paper, a Discrete Event System (DES) model for hybrid control systems is developed in Section 2. This modeling framework is used in Section 3 to develop supervisory controllers. In Section 4, event identification using convex programming methods is discussed.

2 Hybrid Control System Model

A hybrid control system, can be divided into three parts as shown in Figure 1. The models we use for each of these three parts, as well as the way they interact are now described. For a more detailed description and discussion see [13].

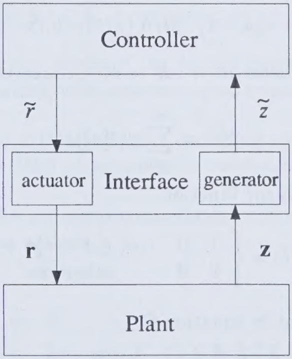


Figure 1: Hybrid Control System

2.1 Plant

The system to be controlled, called the plant, is modeled as a time-invariant, continuous-time system. This part of the hybrid control system contains the entire continuous-time portion of the system, possibly including a continuous-time controller. The plant is represented by

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{r}) \tag{1}$$

where $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{r} \in \mathbb{R}^m$ are the state and input vectors respectively, and $f : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$ is a function.

2.2 Controller

The controller is a discrete event system which is modeled as a deterministic automaton. This automaton can be specified by a quintuple, $\{\tilde{S}, \tilde{Z}, \tilde{R}, \delta, \phi\}$, where $\tilde{S}$ is the (possibly infinite) set of states, $\tilde{Z}$ is the set of *plant symbols*, $\tilde{R}$ is the set of *controller symbols*, $\delta : \tilde{S} \times \tilde{Z} \rightarrow \tilde{S}$ is the state transition function, and $\phi : \tilde{S} \rightarrow \tilde{R}$ is the output function. The controller can be described by the equations

$$\begin{aligned} \tilde{s}[n] &= \delta(\tilde{s}[n-1], \tilde{z}[n]) && (2) \\ \tilde{r}[n] &= \phi(\tilde{s}[n]) && (3) \end{aligned}$$

where $\tilde{s}[n] \in S, \tilde{z}[n] \in \tilde{Z}$, and $\tilde{r}[n] \in \tilde{R}$. The input and output signals associated with the controller are asynchronous sequences of symbols, rather than continuous-time signals. Notice that there is no delay in the controller.

2.3 Interface

An interface is required which can convert continuous-time signals to sequences of symbols and vice versa. The interface consists of two rather simple maps, α and γ.

The *generator* is represented by a function, $\alpha : \mathbb{R}^n \rightarrow \tilde{X}$, and a set of functionals, $h_i : \mathbb{R}^n \rightarrow \tilde{X}$. The null space of each functional forms a separating hypersurface in the state space of the plant and whenever one of these surfaces is crossed a *plant event* occurs. With each plant event, the generator generates the following plant symbol.

$$\tilde{x}[n] = \alpha(\mathbf{x}(\tau[n])) \tag{4}$$

where $\tau[n]$ is the sequence of event instants or times defined as follows.

$$\begin{aligned} \tau[0] &= 0 && (5) \\ \tau[n] &= \inf\{t > \tau[n-1] : \exists i, h_i(\mathbf{x}(t)) \cdot h_i(\mathbf{x}(\tau[n-1] + \epsilon)) < 0\} && (6) \end{aligned}$$

The second map, called the *actuator*, $\gamma : \tilde{R} \rightarrow \mathbb{R}^m$, converts a sequence of controller symbols to a plant input as follows.

$$\mathbf{r}(t) = \sum_{n=0}^{\infty} \gamma(\tilde{r}[n]) I(t) \tag{7}$$

where $I(t)$ is the following indicator function.

$$I(t) = \begin{cases} 1 & \text{if} \quad \tau[n] \leq t < \tau[n+1] \\ 0 & \text{if} \quad \text{otherwise} \end{cases} \tag{8}$$

and $\tau[m]$ is the sequence defined in equation 6.

3 Controllability and DES Controller Design for Hybrid Control Systems

3.1 DES Models for Hybrid Control Systems

If the plant and interface of a hybrid control system are viewed as a single component, this component behaves like a discrete event system. It is advantageous to view a hybrid control system this way because it allows it to be modeled as two interacting discrete event systems which are more easily analyzed than the system in its original form. The discrete event system which models the plant and interface is called the *DES Plant Model.*

In this section we restrict the interface to systems where the plant symbols are associated with the separating hypersurfaces and the plant input is piecewise constant, which allows the DES to be modeled by an automaton. The automaton is specified by a quintuple, $\{\tilde{P}, \tilde{Z}, \tilde{R}, \psi, \xi\}$, where $\tilde{P}$ is the set of states, $\tilde{Z}$ and $\tilde{R}$ are the sets of plant symbols and controller symbols, $\psi : \tilde{P} \times \tilde{Z} \to \tilde{P}$ is the state transition function, and $\xi : \tilde{P} \times \tilde{R} \to \mathbf{P}(\tilde{Z})$ is the enabling function.

The behavior of the DES plant model is as follows

$$\begin{aligned} \tilde{z}[n+1] &\in \xi(\tilde{p}[n], \tilde{r}[n]) && (9) \\ \tilde{p}[n] &= \psi(\tilde{p}[n-1], \tilde{z}[n]) && (10) \end{aligned}$$

where $\tilde{p}[n] \in \tilde{P}, \tilde{r}[n] \in \tilde{R}$, and $\tilde{z}[n] \in \tilde{Z}$. The enabling function defines which events are enabled in a given state and input. The state transition function defines the state which results following the occurrence of an event. The state transition function, ψ, is a partial function because some events are never enabled from a given state. In general, more than one event will be enabled for a given state and input, and therefore the DES plant may be nondeterministic.

If we let $\tilde{Z}^*$ be the set of all strings of plant symbols, then the state transition function can also be used to generate the plant state after a string of events.

$$\tilde{p}[n] = \psi(\tilde{p}[0], \tilde{w}) \tag{11}$$

where $\tilde{w} \in \tilde{Z}^*$ of length n.

The models used in the Ramadge-Wonham framework differ from the above model, and therefore the theorems and techniques from that framework cannot be applied directly, but must be adapted. This is shown in [13] and will not be repeated here.

3.2 Controllability of the Hybrid Control System

For the hybrid control system, the language K is controllable if

$$\forall w \in \bar{K}\ \exists\ \tilde{r} \in \tilde{R}\ \ni\ w\xi(\psi(\tilde{p}_0, w), \tilde{r}) \subset \bar{K} \tag{12}$$

This condition requires that for every prefix of the desired language, K, there exists a control, $\tilde{r}$, which will enable only events which will cause string to remain in $\bar{K}$.

Theorem 1 *If K is prefix closed and controllable according to 12, then a DES controller can be designed which will control the DES plant model to the language K.*

For hybrid control systems, the supremal controllable sublanguage of the DES plant model can be found by a similar iterative scheme.

$$\begin{aligned} K_0 &= K && (13) \\ K_{i+1} &= \{w : w \in \bar{K}, \forall v \in \overline{w}\ \exists\ \tilde{r} \in \tilde{R}\ \ni \\ &\qquad v\xi(\psi(\tilde{p}_0, v), \tilde{r}) \in \overline{K_i}\} && (14) \\ K^{\uparrow} &= \lim_{i \to \infty} K_i && (15) \end{aligned}$$

Theorem 2 *For a DES plant model and language K, $K^{\uparrow}$ is controllable and contains all controllable sublanguages of K.*

With this controllability result we are able to design controllers for hybrid control systems. This is analogous to the controller design in the Ramadge-Wonham framework, and it is not described here.

4 Example - Double Integrator

Using the double integrator example from [14], we have the following plant

$$\dot{\mathrm{x}} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \mathrm{x} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \mathrm{r} \tag{16}$$

In the interface, the function α partitions the state space into four regions as follows,

$$\begin{cases} \tilde{p}_1 & \text{if} \quad x_1, x_2 > 0 \\ \tilde{p}_2 & \text{if} \quad x_1 < 0, x_2 > 0 \\ \tilde{p}_3 & \text{if} \quad x_1, x_2 < 0 \\ \tilde{p}_4 & \text{if} \quad x_1 > 0, x_2 < 0 \end{cases}, \tag{17}$$

and the function γ provides three set points,

$$\gamma(\tilde{r}) = \begin{cases} -10 & \text{if} \quad \tilde{r} = \tilde{r}_1 \\ 0 & \text{if} \quad \tilde{r} = \tilde{r}_2 \\ 10 & \text{if} \quad \tilde{r} = \tilde{r}_3 \end{cases}, \tag{18}$$

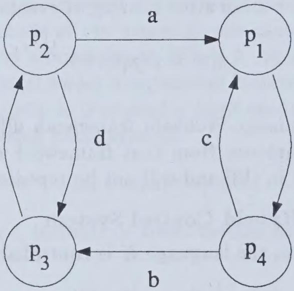


Figure 2: DES Plant Model for Example

Combining the plant and interface, we obtain the DES plant model. This DES is represented by the automaton in figure 2, and explicitly by the following sets and functions.

$$\tilde{P} = \{\tilde{p}_1, \tilde{p}_2, \tilde{p}_3, \tilde{p}_4\} \tag{19}$$

$$\tilde{Z} = \{\tilde{a}, \tilde{b}, \tilde{c}, \tilde{d}\} \tag{20}$$

$$\tilde{R} = \{\tilde{r}_1, \tilde{r}_2, \tilde{r}_3\} \tag{21}$$

$$\psi(\tilde{p}_1, \tilde{d}) = \tilde{p}_4 \quad \psi(\tilde{p}_3, \tilde{c}) = \tilde{p}_2 \tag{22}$$

$$\psi(\tilde{p}_2, \tilde{a}) = \tilde{p}_1 \quad \psi(\tilde{p}_4, \tilde{c}) = \tilde{p}_1 \tag{23}$$

$$\psi(\tilde{p}_2, \tilde{d}) = \tilde{p}_3 \quad \psi(\tilde{p}_4, \tilde{b}) = \tilde{p}_3 \tag{24}$$

$$\xi(\tilde{p}_1, \tilde{r}_1) = \{\tilde{d}\} \quad \xi(\tilde{p}_3, \tilde{r}_3) = \{\tilde{c}\} \tag{25}$$

$$\xi(\tilde{p}_2, \tilde{r}_1) = \{\tilde{a}, \tilde{d}\} \quad \xi(\tilde{p}_4, \tilde{r}_1) = \{\tilde{b}\} \tag{26}$$

$$\xi(\tilde{p}_2, \tilde{r}_2) = \{\tilde{a}\} \quad \xi(\tilde{p}_4, \tilde{r}_2) = \{\tilde{b}\} \tag{27}$$

$$\xi(\tilde{p}_2, \tilde{r}_3) = \{\tilde{a}\} \quad \xi(\tilde{p}_4, \tilde{r}_3) = \{\tilde{b}, \tilde{c}\} \tag{28}$$

The values, for which ψ has not been stated, are undefined, and the values for which ξ has not been stated are the empty set.

The language generated by this automaton is $L = \overline{((dc)^* + db(cd)^*ca)^*}$. If we want to drive the plant in clockwise circles, then the desired language is $K = \overline{(dbca)^*}$. It can be shown that this K satisfies equation (12) and therefore according to theorem 1, a controller can be designed to achieve the stated control goal.

5 Event Identification

The interface is instrumental in determining the *meaning* of the symbols used by the supervisor. Symbols acquire meaning by their association with nonsymbolic entities in the continuous-state plant. For example, a region of the state space or a control policy which successfully transfers a system between two regions of the state space may be used to ground symbols in a meaningful manner. In this section, a set of sufficient conditions are derived which test whether or not the control symbols used by the supervisor are meaningful with respect to their ability for supervising plant behaviour.

This test takes the form of a set of linear inequalities which can be efficiently solved using the method of centers [15]. The method of centers can also be framed as an inductive inference protocol so this method can be realized as an identification algorithm to be used for the on-line verification of a given DES plant automaton model. The significance of these results is that there are some very efficient method of centers algorithms. Prior work in this area has focused on the use of geometric centers such as in the ellipsoid algorithm [16]. While the ellipsoid algorithm is known to be somewhat inefficient, it provides provable results on computational complexity. These results suggest that the *event identification* scheme is learnable in the sense that the number of queries needed to decide on an event binding is bounded in a polynomial manner with plant complexity. Method of center algorithms based on analytic centers should provide faster learnability and the implementation issues associated with these algorithms are currently being investigated. The following two subsections first discuss the notion of supervisability used in this paper and then moves onto methods for solving the resulting inequality tests.

5.1 Supervisability

The combination of plant and interface forms another discrete event system. The hybrid control system, from the supervisor's perspective, can then be seen as two interconnected DES; the supervisor and an equivalent plant DES. Since the supervisor is directly connected to the plant DES, one approach to hybrid controller design is to synthesize a supervisor which effectively controls the equivalent DES model of the plant. The natural concern is whether or not control of the DES is sufficient to "acceptably" control the original plant.

The following discussion indicates precisely what is being controlled by the supervisor. Let $\mathbf{X}$ be a finite open cover of $\Re^n$ consisting of elements $\mathbf{x}_i$, for $i = 1, \ldots, q$. The set of *conjunctive events*, $\mathbf{C}$, generated by $\mathbf{X}$ will consist of all subsets, $\mathbf{c}$, of $\Re^n$ which can be expressed as the intersection of elements in $\mathbf{X}$. In other words, for any conjunctive event $\mathbf{c}$ there exists a collection, I, of integers between 1 and q such that

$$\mathbf{c} = \bigcap_{i \in I} \mathbf{x}_i \tag{29}$$

The set I will be called the index set of the conjunctive event.

The *symbolic* behaviour of the plant is described by the way in which the plant's state transitions between events. Consider a hybrid system with a collection, $\mathbf{C}$, of conjunctive events generated by a finite cover, $\mathbf{X}$, of the state space. Let $\tilde{R}$ be a finite alphabet of control symbols, let $V \subset \mathbf{C}$, and let $A \subset V \times V \times \tilde{R}$. The *plant automaton* associated with the hybrid system is a labeled directed graph, (V, A), where an ordered triple of $(\mathbf{c}_o, \mathbf{c}_t, \tilde{r})$ is in A if and only if the plant's state trajectory, $\bar{x}(t)$ generated by the system equations

$$\frac{d\bar{x}}{dt} = f(\bar{x}, \gamma(\tilde{r})) \tag{30}$$

satisfies the following conditions.

- there exists T_o such that $\bar{x}(t)$ is in the closure of $\mathbf{c}_o$ for $0 \leq t \leq T_o$.
- there exists T_t such that $\bar{x}(t)$ is in the closure of $\mathbf{c}_t$ for $t = T_t$.
- and $\bar{x}(t)$ is not in the closure of $\bigcup_i \mathbf{c}_i$ for $T_o < t < T_t$.

As a model of the plant's symbolic behaviour, the plant automaton provides the basis for determining a supervisor which can control the plant. In order for the resulting supervisor to effectively control the plant between conjunctive events, it is necessary that the plant automaton also be supervisable. The following makes this notion of supervisability more precise. Consider the plant automaton, (V, A) associated with a hybrid control system and consider two events, $\mathbf{c}_o$ and $\mathbf{c}_t$, which are elements of V. Let $\tilde{r}[n]$ denote the control history for a directed walk from vertex $\mathbf{c}_o$ to $\mathbf{c}_t$. The event $\mathbf{c}_t$ is said to be *supervisable from* $\mathbf{c}_o$ if and only if any other directed walk originating from $\mathbf{c}_o$ with control history $\tilde{r}[n]$ necessarily terminates at $\mathbf{c}_t$.

The following proposition provides a sufficient condition for testing whether or not an arc of a plant automaton is indeed supervisable

Proposition 1 *Consider a hybrid system with an associated basis event collection,* $\mathbf{X}$. *Let* (V, A) *be a given plant automaton. Let* $V_i : \Re^n \rightarrow \Re$ *(i=1,...,q) be a family of* C^1 *functionals and let* $f_j : \Re^n \rightarrow \Re^n$ $(j = 0, \ldots, m)$ *be a family of Lipschitz continuous control policies. Let* $L_j V_i$ *be the Lie derivative of the functional* V_i *with respect to vector field* f_j. *The event* $\mathbf{c}$ *in* $\mathbf{C}$ *with index set* I *will be supervisable if*

- $V_i(\bar{x}) > 0$ *for all* $\bar{x} \notin \mathbf{c}_i$ *and* $V_i(\bar{x}) = 0$ *for all* $\bar{x} \in \mathbf{c}_i$.
- *for all* $i \in I$ *and for all* $\bar{x} \in \Re^n$,

$$0 > \begin{pmatrix} L_0 V_i & L_1 V_i & \cdots & L_m V_i \end{pmatrix} \begin{pmatrix} 1 \\ r_1 \\ \vdots \\ r_m \end{pmatrix} \tag{31}$$

- *and for all* $\bar{x}$ *on the boundary of events* $\mathbf{c}_o \neq \mathbf{c}$, *with index sets* I_o, *there exists an* $i \in I_o$ *such that* $V_i(\bar{x}) = 0$ *and*

$$0 < \begin{pmatrix} L_0 V_i & L_1 V_i & \cdots & L_m V_i \end{pmatrix} \begin{pmatrix} 1 \\ r_1 \\ \vdots \\ r_m \end{pmatrix} \tag{32}$$

Proof: (outline) In order for event $\mathbf{c}$ to be supervisable there must be an arc which goes from every event in V to $\mathbf{c}$ such that the state trajectory does not intersect any other event in V. One way to insure that this occurs is to require that $\mathbf{c}$ be a globally attracting invariant set and that all other events are repellors of the plant controlled by $\gamma(\tilde{r})$. The above conditions are based in a straightforward manner on the LaSalle invariance principle. •

Remark: If the set of conjunctive events form a full partition of the plant's state space, then supervisability (determinism) is too restrictive to be useful. Insuring determinism over full state partitions will require that the partition boundaries be integral manifolds of the system; a condition which will probably be impossible to satisfy exactly in practice. The formulation stated above, however, does not require that events fully partition the state space. Therefore it is no longer necessary that event boundaries be integral manifolds. Consequently, there is significant flexibility in the way we can assign the control vectors $\bar{r}$ to insure determinism in the plant automaton.

Remark: The inequalities used to test the plant automaton's supervisability require that the Lie derivatives of the dynamic system be known. If there exist prior models for the plant, then these Lie derivatives can be estimated directly from state estimates. If no such models exist however, then these derivatives must be estimated by direct observation of the system. In this respect, the preceding inequalities indicate precisely what type of information must be made available to the system so that it can determine whether or not its plant automaton is indeed supervisable.

Remark: The conditions stated above are global statements. In many situations, however, satisfaction of these conditions on a local basis may be extremely useful. For example, if the plant's state trajectory is known to be contained within a given region of the state space, then it is only necessary that the above inequalities be satisfied in that region (rather than the entire state space) to insure the supervisability of the resulting plant automaton.

5.2 Identification

The sufficient conditions obtained in the preceding section form a system of inequalities which are linear in the control vector, $\bar{r}$. Since the control vector is determined by the hybrid system's actuator mapping, these conditions also provide a method for determining an actuator mapping, γ which insures that the given automaton will be implementable. Such γ (or rather the $\bar{r}$ associated with the control symbol $\tilde{r}$) represent feasible points of the inequality system. The identification of those $\bar{r}$ which yield a supervisable plant automaton can be viewed as an event identification problem. In this case, the event is the arc between two conjunctive events. The identification algorithm determines whether or not such an arc can be actually realized by the system.

The event identification problem is actually a problem in determining feasible points for inequality systems. There exist good numerical techniques for finding such points. One class of algorithms for determining such points uses the so-called method of centers to update a hypothesized feasible point. The method of centers computes a sequence of convex bodies and their centers in such a way that the computed centers converge to the feasible point. Depending on the analytic form of the convex bodies and the centers, different types of algorithms are obtained. Examples of such algorithms are the so-called ellipsoid method [17] and interior point methods based on logarithmic barrier functions [18].

The method of centers can be viewed as an inductive inference algorithm. Inductive inference is a machine learning algorithm in which a system learns a Boolean function from examples[19]. The learning algorithm is outlined below.

- Hypothesis: The hypothesis assumes that the conjunctive event **c** is supervisable by vector $\bar{r}_i = \gamma(\tilde{r}_i)$. This hypothesis is represented by a convex body K_i in which $\bar{r}_i$ is a "center" for K_i. Prior work [16] used ellipsoidal convex bodies and their associated geometric centers. It is quite possible and probably desirable to also use analtyic centers defined by logarithmic barrier functions. This latter approach has proven useful in interior point linear programming algorithms [18].
- Experiment: The inequalities in the proposition indicate that specific information has to

be gathered by the system before the sufficient conditions can be tested. The experimental component of the algorithm involves the measurement of these quantities. In particular, the system must be able to estimate or measure the Lie derivatives of the plant. As noted above, these derivatives can be estimated if there is a prior plant model. When no such model is available, there must be sensor capable of measuring or estimating these derivatives.

- Query: The third component of the algorithm is the oracle query. The oracle is a Boolean functional which outputs 0 if the experimental data is consistent with the inequality system and which outputs 1 if the data does not satisfy the inequality system.

- Update: If the oracle's response is 0, then nothing is done. If the oracle declares 1, then the current data is inconsistent with the hypothesis that $\bar{r}_i$ satisfies the inequality system. This means that the hypothesis has to be changed. An update algorithm is called to compute a new convex body K_i and its associated center. The center of the updated convex body is then taken to be the new control vector $\bar{r}_i$. Prior work examined the use of central cut ellipsoid algorithms [16] in performing this update step. More recent experimental work has implied that the use of analytic centers (rather than geometric centers) may provide a more efficient way of updating the hypothesis.

The advantage of using the method of centers in solving such problems is that these methods have provable polynomial complexity. There are well-known results on the convergence of such methods. For example, an idealized method of centers algorithm which uses the geometric center can be shown to converge in finite time [15]. It is further shown that this finite convergence time scales in a way which is independent of problem complexity. The problem with the idealized algorithm is that the computation of geometric centers for arbitrary convex bodies is impractical. The central-cut ellipsoid algorithm represents a more practical way of computing geometric centers. For this algorithm, it has been shown that the finite convergence time scales on the order of $o(n \ln L)$ where n is a measure of plant complexity (generally the number of control policies available to control the plant) and L is a measure of the volume of the feasible set that's being searched for [20] [16].

The significance of the preceding discussion is that it suggests that the controllability of individual events in the plant automaton can be decided quickly and efficiently using the method of centers. The scaling results suggest that the $o(n \ln L)$ convergence rate for the ellipsoid method can be improved upon considerably and that from a theoretical perspective this rate will be bounded below in a manner which is independent of plant complexity. This immediately suggests that more efficient approaches than the ellipsoid method can be used for finding feasible points. This fact is already well-known and interior point algorithms using analytic centers based on so-called logarithmic barrier functions [18] represent an important class of update procedures which may provide even more efficient identification of supervisable automaton events.

Acknowledgement - the partial support of NSF grant MSS-9216559 is acknowledged.

References

[1] J. Albus, et al, "Theory and practice of intelligent control", In *Proc. 23rd IEEE COMPCON*, pp. 19–39, 1981.

[2] P. J. Antsaklis, K. M. Passino, and S. J. Wang, "Towards intelligent autonomous control systems: Architecture and fundamental issues", *Journal of Intelligent and Robotic Systems*, vol. 1, pp. 315–342, 1989.

[3] P. J. Antsaklis, K. M. Passino, and S. J. Wang, "An introduction to autonomous control systems", *IEEE Control Systems Magazine*, vol. 11, no. 4, pp. 5–13, June 1991.

[4] P. J. Antsaklis and K. M. Passino, "Introduction to intelligent control systems with high degree of autonomy", In *An Introduction to Intelligent and Autonomous Control*, edited by P. J. Antsaklis and K. M. Passino, chapter 1, pp. 1–26. Kluwer Academic Publishers, 1993.

[5] P. J. Antsaklis and K. M. Passino, editors, *An Introduction to Intelligent and Autonomous Control*, Kluwer Academic Publishers, 1993, 448 p.

[6] "Special issue on autonomous intelligent machines", *IEEE Computer*, vol. 22, no. 6, June 1989.

[7] K. M. Passino and P. J. Antsaklis, "Modeling and analysis of artificially intelligent planning systems", In *Introduction to Intelligent and Autonomous Control*, edited by P.J.Antsaklis and K.M.Passino, chapter 8, pp. 191–214. Kluwer, 1993.

[8] G. N. Saridis, "Toward the realization of intelligent controls", *Proceedings of the IEEE*, vol. 67, no. 8, pp. 1115–1133, Aug. 1979.

[9] G. N. Saridis, "Foundations of the theory of intelligent controls", In *Proceedings of the IEEE Workshop on Intelligent Control*, pp. 23–28, 1985.

[10] G. N. Saridis, "Knowledge implementation: Structures of intelligent control systems", In *Proceedings of the IEEE International Symposium on Intelligent Control*, pp. 9–17, 1987.

[11] G. N. Saridis, "Analytic formulation of the principle of increasing precision with decreasing intelligence for intelligent machines", *Automatica*, vol. 25, no. 3, pp. 461–467, 1989.

[12] B. P. Zeigler, *Multifacetted Modelling and Discrete Event Simulation*, Acedemic Press, NY, 1984.

[13] J. A. Stiver and P. J. Antsaklis, "DES supervisor design for hybrid control systems", In *Proceedings of the Twenty-Ninth Annual Allerton Conference on Communication, Control, and Computing*, University of Illinois at Urbana-Champaign, Sep. 1993, To appear.

[14] J. A. Stiver and P. J. Antsaklis, "Modeling and analysis of hybrid control systems", In *Proceedings of the 31st Conference on Decision and Control*, pp. 3748–3751, Tucson, AZ, Dec. 1992.

[15] A.S. Nemirovsky and D.B. Yudin, *Problem Complexity and Method Efficiency in Optimization*, Wiley and Sons, 1983.

[16] M. D. Lemmon, J. A. Stiver, and P. J. Antsaklis, "Learning to coordinate control policies of hybrid systems", In *Proceedings of the American Control Conference*, pp. 31–35, San Francisco, CA, June 1993.

[17] R.G. Bland, D. Goldfarb, and M.J. Todd, "The ellipsoid method: a survey", *Operations Research*, vol. 29, pp. 1039–1091, 1981.

[18] C.C. Gonzaga, "Path-following methods for linear programming", *SIAM Review*, vol. 34, no. 2, pp. 167–224, June 1992.

[19] D. Angluin and C.H. Smith, "Inductive inference: Theory and methods", *Computing Surveys*, vol. 15, no. 3, pp. 237–269, September 1983.

[20] Groetshel, Lovasz, and Schrijver, *Geometric Algorithms and Combinatorial Optimization*, Springer-Verlag, Berlin, 1988.

Diagnosing Process Faults Using Neural Network Models

Kevin L. Buescher, Roger D. Jones, and Michael J. Messina*
Los Alamos National Laboratory
Mail Stop F645
Los Alamos, NM 87544
email to: klb@lanl.gov

Abstract

In order to be of use for realistic problems, a fault diagnosis method should have the following three features. First, it should apply to nonlinear processes. Second, it should not rely on extensive amounts of data regarding previous faults. Lastly, it should detect faults promptly. We present such a scheme for static (i.e., non-dynamic) systems. It involves using a neural network to create an associative memory whose fixed points represent the normal behavior of the system.

1 Introduction

The more stringent constraints placed on processes due to concerns over efficiency, safety, and environmental factors increase the need for process fault diagnosis. There exists a great deal of literature regarding fault diagnosis (see [1], [2], [3], [4], and [5]), but much of this has limited use because the assumptions made about the underlying process or the availability of data are generally not satisfied in practice. In this paper, we attempt to narrow the gap between existing fault diagnosis schemes and what is needed for real processes.

We distinguish between three levels of fault diagnosis. Fault *detection* is simply deciding whether or not a fault is present. Fault *classification* entails determining what kind of fault has occurred (e.g., which sensor or actuator failed?). Fault *estimation* involves determining the extent of the fault (e.g., what is the difference between the faulty sensor's reading and the actual value?).

There are three properties that are desirable for a practical fault diagnosis scheme, but these are rarely *all* present in a given method. First, the scheme should apply to nonlinear processes. Most techniques assume that the underlying process is linear. Such an assumption may suffice when the objective is to control the process. However, fault diagnosis schemes typically rely on a close match between the model and the process, so the assumption of linearity it harder to justify. (There are, however, some results pertaining to *robust* fault detection [6].)

---

*Please address all correspondence to the first author. The research reported here has been supported by the United States Department of Energy.

Second, a fault diagnosis scheme should not rely on first-principles models or extensive data from previous faults; empirical data from the normal operation of the process should suffice. Some methods consist of pattern recognition techniques applied to a database of faulty operating conditions (see [7], [8], and [9]). Such data may be too expensive or dangerous to collect. Also, in some industries which operate a number of diverse processes (notably, the chemical processing industry), obtaining first-principles models is not economically feasible.

Lastly, a fault diagnosis scheme should detect faults promptly, and not wait to generate a warning until an operating condition is violated. If a process is being controlled, the controller may compensate for incipient faults. However, if the process is stopped for maintenance purposes, it is important know what noncritical faults are occurring (e.g., a sensor is drifting) so that these can be repaired.

We now discuss our approach to fault diagnosis and note how it relates to other methods. Many fault detection schemes are based on learning the normal behavior of a process and then detecting differences between this and its current behavior. This idea of using a model of the process to relate variables in order to check for inconsistencies is know as *analytical redundancy.* For instance, if there are two variables of interest in a process, x, y, we may find that they satisfy a relation $r(x, y) = 0$ during normal operation. If we make measurements $\hat{x}, \hat{y}$, we can compute $r(\hat{x}, \hat{y})$ and use its value to decide if the process is behaving normally. Quantities such as r are known as *parity* equations and $r(\hat{x}, \hat{y})$ is treated as a residual which is used in statistical decision tests to determine if a fault is present.

The drawback to examining parity equations is that the residuals $r(\hat{x}, \hat{y})$ are not always directly helpful in classifying faults. Our approach is to find values $\bar{x}$ and $\bar{y}$ such that $r(\bar{x}, \hat{y}) = 0$ and $r(\hat{x}, \bar{y}) = 0$ in order to determine what possible changes in x and y could have caused $r(\hat{x}, \hat{y}) \neq 0$. The residuals $\hat{x} - \bar{x}$ and $\hat{y} - \bar{y}$ are well-suited to finding faults in x and y.

We call $\bar{x}$ and $\bar{y}$ *inferred* values of x and y. Essentially, finding them involves solving $r(x, y) = 0$ in terms of x and y. The difficulty lies in cases where r is multiple-valued in x or y. We have addressed this problem by developing an iterative technique (based on a neural network) which, when inferring the value of x, normally finds the closest $\bar{x}$ to $\hat{x}$

such that $r(\bar{x}, \hat{y}) = 0$. In this sense, our technique functions as an associative memory, with the surface $r(x, y) = 0$ as the fixed points.

Our method only applies to static processes (the intended application is processes that are operated in a steady state). Further, we will restrict our attention to the case where at most one fault is present. We focus on the process of generating residuals and do not carry out statistical testing.

In Section 2, we describe our method for inferring the value of variables and present some simulation results. We also show how this is useful for fault detection. In Section 3, we show how this capability is instrumental for fault classification and estimation.

2 Fault Detection

The ability to infer the value of one variable from the value of others is the key to our fault diagnosis scheme. An example best illustrates this approach.

Suppose that there are two variables of interest in our system, x and y, and that they are related as follows:

$$y = 4x(1-x), \ 0 \leq x \leq 1. \tag{2.1}$$

Let us denote the measured values of x and y by $\hat{x}$, $\hat{y}$ and their true values by x^*, y^*. One way to detect a fault is to examine the quantity

$$P = \hat{y} - 4\hat{x}(1-\hat{x}), \tag{2.2}$$

which should be zero in the absence of noise and faults. A relation such as (2.2) is known as a *parity equation.* Relations such as this can be used to generate residuals for the purpose of fault detection (see [10] and [5]). However, we are interested in other quantities which can also be used for fault *classification.*

Our approach is to solve (2.1) for both x and y:

$$\bar{y} = 4\hat{x}(1-\hat{x}) \tag{2.3}$$

and

$$\bar{x} = \begin{cases} (1+\sqrt{1-\hat{y}})/2, & \hat{x} > 1/2 \\ (1-\sqrt{1-\hat{y}})/2, & \text{otherwise} \end{cases}. \tag{2.4}$$

We say that the quantities $\bar{x}$ and $\bar{y}$ are *inferred* from the measured values. Note that, of the two possible values for $\bar{x}$ given $\hat{x}$, (2.4) specifies the one closest to $\hat{x}$. Thus, if $\hat{y} = y^*$ and $\hat{x}$ is a slightly perturbed value of x^*, we will have $\bar{x} = x^*$. In the absence of noise and faults, the residuals

$$\delta_x = (\hat{x} - \bar{x}) \text{ and } \delta_y = (\hat{y} - \bar{y})$$

will be zero. These quantities can be fed into a statistical decision procedure to detect faults. In the next section, we describe how to classify faults using inferred values.

We now turn to the problem of how to approximate relations such as (2.3) and (2.4) from data points (x_i, y_i) gathered during normal operation of the process (i.e., (x_i, y_i) that satisfy (2.1)). The difficulty lies in approximating relations such as (2.4) that have a multiple-valued nature.

In this example, we could easily determine (2.3) from data and then solve it for x to arrive at a relation such as (2.4). However, this will not work in a case where the true relation between x and y is multiple-valued in each variable (e.g., $x^2 + y^2 = 1$).

We can solve this problem by finding a function f such that, upon iterating

$$x(n+1) = f(x(n), \hat{y})$$

with the initial condition $x(0) = \hat{x}$, $x(n)$ approaches $\bar{x}$, where $\bar{x}$ is the closest point to $\hat{x}$ such that $(\bar{x}, \hat{y})$ satisfies (2.1). Our approach to finding such an f is to train a neural network using two types of data. The first type of data is simply data from the normal operation of the process; the network (f) is trained to perform the mapping

$$(x_i, y_i) \mapsto x_i, \tag{2.5}$$

where (x_i, y_i) satisfies (2.1). The second type of data is generated from the (x_i, y_i). The network is taught to perform the mapping

$$(x_i + \eta_i, y_i) \mapsto x_i, \tag{2.6}$$

where the η_i are small, zero-mean random numbers. The effect of (2.5) is to make points (x, y) that satisfy (2.1) fixed points of the iteration $x(n+1) = f(x(n), y)$. The effect of (2.6) is to shape the dynamics of this iteration so that the parabola (2.1) attracts points near it.

Figure 1 shows the results from a neural network trained in this manner. (The neural network is described in the Appendix.) The training points consisted of 500 points randomly sampled from the parabola and 500 additional points created as in (2.6) with each η_i drawn uniformly from $[-0.025, -0.025]$. Figure 1 shows the time histories of 19 randomly drawn initial points. They do in fact converge to points very near the parabola. Only the point near the apex of the parabola failed to converge to the closest side. This figure is typical. Except for a small strip about $x = 1/2$ and a small region about the apex of the parabola, all points usually converge to the nearest side of the parabola, as desired.

In Figure 2, the initial training data was corrupted by noise as follows. Values of x were chosen randomly. A random number drawn uniformly from $[-0.05, 0.05]$ was then added to the corresponding value of y. The performance has not degraded significantly.

We have performed experiments involving relations other than (2.1) and found that the results above are typical.

The preceding discussion pertains to a specific example. Let us summarize it in a more general context. Suppose that there are three variables, w, x, and y, in a process that satisfy a relation $r(w, x, y) = 0$ under normal operating conditions, and that you have a number of data points (w_i, x_i, y_i) that satisfy $r(w_i, x_i, y_i) = 0$. To create a function (call it h) that can be used to infer the value of w, train a neural network to perform the mappings $(w_i, x_i, y_i) \mapsto w_i$ as well as the mappings $(w_i + \eta_i, x_i, y_i) \mapsto w_i$, where the η_i are small, zero-mean random numbers. A separate network can be trained for each of the three variables.

To infer the value of w from a measurement $(\hat{w}, \hat{x}, \hat{y})$, iterate $w(n+1) = h(w(n), \hat{x}, \hat{y})$ from the initial condition $w(0) = \hat{w}$. If $w(n)$ converges to a value $\bar{w}$, declare this as the inferred value of w. If the measured values are such that $r(\hat{w}, \hat{x}, \hat{y}) \neq 0$, then the residual $\delta_w = \hat{w} - \bar{w}$ should be nonzero. To decide whether a fault is present, a statistical test can be performed to check if the residuals δ_w, δ_x, and δ_y are significantly different from zero.

3 Fault Classification and Estimation

In this section, we briefly describe two techniques for fault classification and estimation that rely on the ability to infer the value of one variable from the others. It is impossible to classify a fault based on a *single* measurement from a *single* relation such as (2.1). Accordingly, the first classification method we describe involves more than one relation, and the second involves more than one measurement. Both of these methods are intended to work under the assumption that only a *single fault* has occurred.

3.1 Analytical redundancy

The idea underlying this approach is that if we know two or more independent relations involving a variable of interest, we can use this extra information to determine whether a variable is faulty. (See [10], [4] and [5] for discussions of analytical redundancy. Our approach is inspired by [11].) We will use an example to illustrate this.

Suppose that, in addition to satisfying

$$y = 4x(1 - x), \tag{3.1}$$

the variables x and y also satisfy

$$z = x + y, \tag{3.2}$$

where z is a quantity that is considered to be immune to faults (e.g., z is the result of a lab test). Let $\hat{x}$ and $\hat{y}$ be the measured values of x, y and x^*, y^* be their true values. By assumption, $z = x^* + y^*$. Also, let $\bar{x}_1, \bar{y}_1$ denote values of x, y inferred from (3.1) and $\bar{x}_2, \bar{y}_2$ denote values inferred from (3.2) (e.g., $\bar{x}_2 = z - \hat{y}$). In practice, we would use the method of the preceeding section to determine the inferred values.

If no faults occur, the inferred values will match the measured values (ignoring noise). If we know that *only one* fault has occurred, we can usually classify it by noting that if the inferred values for a given variable are all alike, this provides very strong evidence that this variable is faulty. In fact, in this example the only time when $\bar{y}_1 = \bar{y}_2$ and x, *not* y, is faulty occurs when

$$(x^* - \hat{x}) = (8x^* - 5)/4, \tag{3.3}$$

a very special case. However, in this case, we would also have $\bar{x}_1 = \bar{x}_2 \neq \hat{x}$, which would alert us that we cannot classify this fault. Thus, if only one fault has occurred,

$$(\bar{y}_1 = \bar{y}_2 \neq \hat{y} \text{ and } \bar{x}_1 \neq \bar{x}_2) \Rightarrow \;\; y \text{ is faulty.}$$

An analogous condition holds for detecting faults in x. Note that, having classified the fault as being in, say, the variable x, we can estimate the extent of the fault as simply $\hat{x} - \bar{x}$.

The approach discussed above can be extended to classifying single faults in more than two variables. All that is required is that each variable appear in at least two relations.

3.2 Fault models

The idea behind this approach is simple. Suppose that you obtained a sequence of measurements $(\hat{x}_i, \hat{y}_i)$ from the system (3.1) (after obtaining the initial fault-free data, (x_i, y_i)) and plotted the measurements. If the plot resembled a parabola shifted along the x-axis with respect to (3.1), you would probably conclude that the variable x was faulted.

The key requirement is that you should know in advance what form a fault will take. For instance, if x and y are temperature sensors, a good fault model might be

$$\hat{x} = a_x x^* + b_x, \;\; \hat{y} = a_y y^* + b_y.$$

That is, the sensor may be experiencing an offset ($b \neq 0$) or a gain fault ($a \neq 1$).

Now suppose that you have gathered measurements $(\hat{x}_i, \hat{y}_i)$ for which x, but not y, is faulty. If the faults are not too extensive, the corresponding inferred values $(\bar{x}_i, \bar{y}_i)$ would be such that each $\bar{x}_i$ is close to the true value, x^*. Thus, there would exist a_x and b_x

that fit $\hat{x}_i = a_x \bar{x}_i + b_x$ well for each i. Unless the underlying relation between x and y is linear, it is unlikely that a good fit could be obtained for $\hat{y}_i = a_y \bar{y}_i + b_y$. Not only does this classify the fault, but it allows us to estimate its extent as well.

4 Concluding Remarks

We have presented some ideas for fault diagnosis that meet our criteria with respect to nonlinearity of the process, using empirical, fault-free data, and detecting incipient faults. Of course, the true test of these ideas will be to use data from a real process and to carry out the required statistical tests to diagnose faults. We would like to extend these ideas to dynamic systems and the detection of multiple faults.

Appendix: The Underlying Neural Network

The neural network used to produce Figures 1 and 2 is a type of radial basis function known as a CNLS[1] network. It is described in detail in [12] and [13], and only a brief description will be given here. A CNLS network with m inputs and n nodes takes the form

$$f(\vec{x}) = \frac{\sum_{i=1}^{n} (a_i + (\vec{x} - \vec{c}_i)' \vec{b}_i) \rho_i(\vec{x})}{\sum_{j=1}^{n} \rho_j(\vec{x})},$$

where $\rho_i(\vec{x}) = \exp(-\beta_i (\vec{x} - \vec{c}_i)'(\vec{x} - \vec{c}_i))$. Here, $\vec{x}$, $\vec{c}_i$, and $\vec{b}_i$ are members of $I\!R^m$, and $a_i, \beta_i \in I\!R$. The adjustable parameters are a_i, $\vec{b}_i$, $\vec{c}_i$, and β_i. The ρ_i are "basis functions" centered at $\vec{c}_i$.

Typically, these networks are trained by fixing values for the $\vec{c}_i$ and β_i and adjusting the a_i and $\vec{b}_i$ using a least-squares procedure. Usually, the $\vec{c}_i$ are randomly drawn from the data points. Since f is linear in a_i and $\vec{b}_i$, there are no local minima for a sum-of-squares error criterion and the training is quite fast. The CNLS networks appears to be relatively insensitive to the values of n and the β_i.

The network used to produce Figures 1 and 2 had $n = 25$ and $\beta_i \equiv \beta = 50$.

References

[1] A. S. Willsky, "A survey of design methods for failure detection in dynamic systems," *Automatica*, vol. 12, pp. 601–611, 1976.

[2] R. Isermann, "Process fault detection based on modeling and estimation methods–a survey," *Automatica*, vol. 20, pp. 387–404, 1984.

---

[1] Connectionist Normalized Linear Spline

[3] M. Basseville and A. Benveniste, eds., *Detection of Abrupt Changes in Signals and Dynamical Systems*, vol. 77 of *Lecture Notes in Control and Information Sciences.* Berlin: Springer-Verlag, 1986.

[4] R. Patton, P. Frank, and R. Clark, eds., *Fault Diagnosis in Dynamical Systems: Theory and Applications.* New York: Prentice Hall, 1989.

[5] P. M. Frank, "Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy–a survey and some new results," *Automatica*, vol. 26, no. 3, pp. 459–474, 1990.

[6] X.-C. Lou, A. S. Willsky, and G. C. Verghese, "Optimally robust redundancy relations for failure detection in uncertain systems," *Automatica*, vol. 22, no. 3, pp. 333–344, 1986.

[7] J. C. Hoskins, K. M. Kaliyur, and D. M. Himmelblau, "Fault diagnosis in complex chemical plants using artificial neural networks," *AIChE Journal*, vol. 37, no. 1, pp. 137–141, 1991.

[8] T. Sorsa and H. N. Koivo, "Application of artificial neural networks in process fault diagnosis," *Automatica*, vol. 29, no. 4, pp. 843–849, 1993.

[9] J. A. Leonard and M. A. Kramer, "Radial basis function networks for classifying process faults," *IEEE Control Systems Magazine*, vol. 11, no. 3, pp. 31–38, 1991.

[10] E. Y. Chow and A. S. Willsky, "Analytical redundancy and the design of robust failure detection systems," *IEEE Transactions on Automatic Control*, vol. AC-29, no. 7, pp. 603–614, 1984.

[11] S. Saxena and A. Unruh, "Diagnosis of semiconductor manufacturing equipment and processes." Submitted to *IEEE Transactions on Semiconductor Manufacturing*, 1993.

[12] R. D. Jones *et al.*, "Nonlinear adaptive networks: A little theory, a few applications," in *Proceedings of the First Los Alamos Workshop on Cognitive Modeling in System Control: Theoretical Foundations and Prospects for Applications*, (Sante Fe, NM), June 1990.

[13] R. D. Jones, Y. Lee, C. W. Barnes, G. W. Flake, K. Lee, P. S. Lewis, and S. Qian, "Function approximation and time series prediction with neural networks," in *Proceedings of the International Joint Conference on Neural Networks*, (San Diego, CA), pp. I–649–666, June 1990.

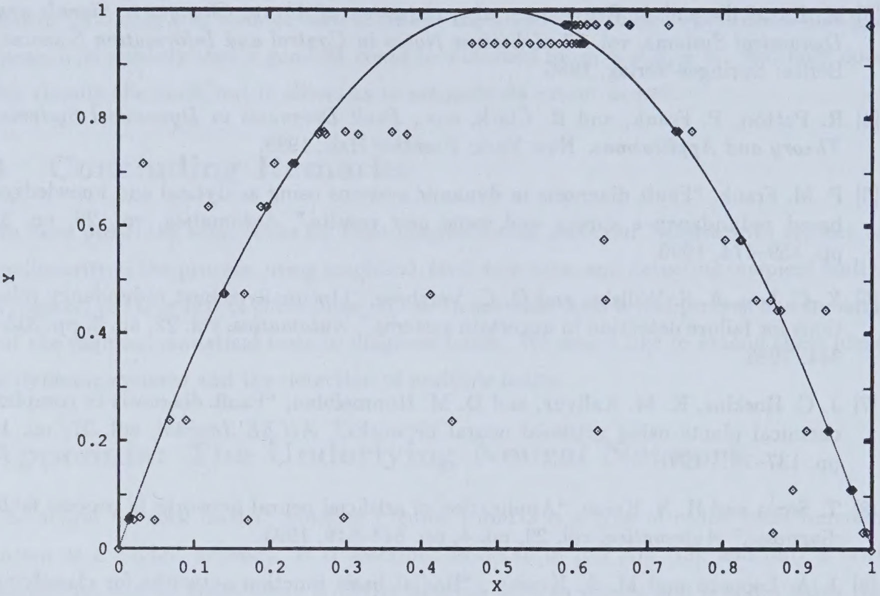


Figure 1. Convergence of iterations, clean data.

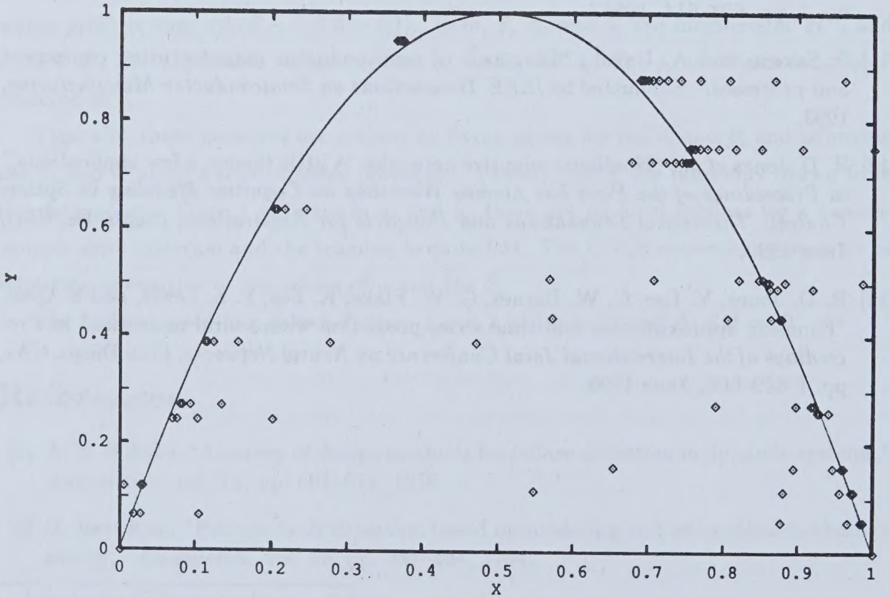


Figure 2. Convergence of iterations, noisy data.

On Bandit Problems With Side Observations and Learnability

Sanjeev R. Kulkarni*

Department of Electrical Engineering
Princeton University, Princeton, N.J., 08544
kulkarni@ee.princeton.edu

Abstract

We introduce a general class of bandit-type problems and discuss connections with results in learning theory. Our model is formulated as follows. We assume that two unknown conditional distributions $P_1(y|x), P_2(y|x)$ on $\mathbb{R}$ given $x \in X$ are selected from some known classes of distributions. We observe sequentially a set of points $x_1, x_2, \ldots$. On observing x_n we must select from which of the two distributions to obtain a reward. Our goal is to maximize the expected sum of the rewards. This model generalizes the classical two-armed bandit problem by allowing the decision at each time to depend on a side observation. The model combines bandit-type problems with non-parametric regression estimation. We use results from learning and statistical inference to obtain bounds on the achievable regret for such problems.

1 Introduction

The classical two-armed bandit problem is perhaps the simplest model which captures a variety of aspects (such as the standard identification versus control or exploration versus exploitation tradeoffs) inherent in many learning and adaptive systems. Bandit problems are important in sequential design and adaptive control, and have received much attention since the early papers of Thompson [13] and Robbins [12] (for example, see the books [5, 7] for various perspectives).

In this paper, we introduce a general class of bandit-type problems formulated as follows. We assume that there are two arms characterized by unknown conditional probability distributions $P_1(y|x)$ and $P_2(y|x)$. We obtain sequentially a set of points $x_1, x_2, \ldots$. On receiving x_n we must select from which of the two conditional distributions to observe a reward y_n. Our goal is to maximize the expected sum of the rewards. The problem considered here is more general than the setup of standard bandit problems, with a very special case giving the classical two-armed bandit problem. We use results from learning and statistical inference to obtain bounds on the achievable regret for such problems. Thus, the results also provide a link between bandit-type problems and problems in non-parametric regression estimation studied in machine learning and pattern recognition.

Woodroofe [17] has also considered a model allowing a side observation x_n, which he called a concomitant variable, although the problem we consider here is somewhat different from that in [17]. Our formulation is quite a bit more general in that we consider

---

*This work was supported in part by the National Science Foundation under grant IRI-9209577 and by the U.S. Army Research Office under grant DAAL03-92-G-0320.

a two-armed problem in which the conditional distribution on each arm is allowed to come from a general class of distributions. Our formulation is also different in that we do not use a Bayesian formulation. Our results are more in the spirit of the results of [9] and the subsequent work in [1, 2, 3, 4]. The formulation presented here is more general in that we allow side observations, but we do not have tight bounds. At present, our results provide only upper bounds on the regret, and we believe these bounds can certainly be improved. However, the bounds hold for a very broad range of problems and require few regularity assumptions. We believe the formulation and results are an interesting starting point for further work.

2 Background and Formulation

In the classical two-armed bandit problem (originating from the work in [12, 13]), there are two unknown distributions P_1 and P_2 associated with arm 1 and arm 2, respectively. At each time we are allowed to select an arm from which to receive a reward drawn according to the distribution for that arm. Our goal is to maximize the expected sum of the rewards. A great deal of work has been done on this problem and various generalizations (e.g., see [5, 7] and the references therein). There are several standard variations such as multiple arms, various types of discounting of the rewards, Bayesian settings, etc., and much of the work is devoted to studying properties of particular strategies and obtaining bounds on the achievable regret under various setting.

One important result was obtained relatively recently by Lai and Robbins [9] in which they characterized the optimum performance achievable in a multi-armed bandit problem. Specifically, suppose there are k arms with unknown distributions $P_1, \ldots, P_k$ and corresponding means $\mu_1, \ldots, \mu_k$. Let μ^* be the maximum mean among $\mu_1, \ldots, \mu_k$. If the distributions on the arms were known, then the optimum decision at each time in order to maximize the expected reward would be simply to always select the arm which has the maximum mean μ^*. However, since the actual distributions on the arms are unknown, an algorithm must make a selection based on past choices and observations. Let ϕ_n denote the arm selected at time n. Then the reward y_n received at time n is drawn according to the distribution P_{ϕ_n} and so has expected value μ_{ϕ_n}. The cumulative reward received up to and including time n is denoted S_n and its expected value is denoted ES_n. The goal is to maximize ES_n.

A natural quantity to measure the performance of a decision rule ϕ is the regret $R_n = R_n(P_1, \ldots, P_k)$ defined as the difference between the expected reward of the optimal decision rule and the expected reward of the decision rule ϕ. Namely,

$$R_n = n\mu^* - ES_n \tag{1}$$

The rate of growth of R_n indicates how well the decision rule ϕ is performing asymptotically, with a slower rate of growth indicating better performance. Lai and Robbins [9] showed essentially that slowest achievable rate of growth is $O(\log n)$. Specifically, under suitable regularity conditions on the distributions of the arms, they showed that if ϕ is a decision rule such that for every set of distributions $P_1, \ldots, P_k$ the regret satisfies $R_n(P_1, \ldots, P_k) = o(n^a)$ for every $a > 0$ then for every set of distributions such that the μ_j are not equal

$$\liminf_{n\to\infty} R_n(P_1, \ldots, P_k)/\log n \geq \sum_{j:\mu_j<\mu^*} (\mu^* - \mu_j)/I(P_j, P^*) \tag{2}$$

where P^* is the distribution corresponding to the arm with mean μ^* and $I(P_j, P^*)$ denotes the relative entropy between P_j and P^*. Moreover, they exhibited a decision rule which

which achieves equality in equation (2). Using the techniques of Lai and Robbins [9], a number of extensions to this result for situations with multiple plays, switching costs, etc., were obtained by Anantharam et al. [3, 4] and Agrawal et al. [1, 2].

We consider a generalized bandit problem, in which we observe an auxiliary variable, and can make a selection of the arms based on this side observation. We consider only the case of two arms, although a similar model could easily be formulated for multiple arms. We will use a measure of regret analogous to that of equation (1), but suitably modified to accommodate the more general formulation. We now turn to a formal description of the problem considered in this paper.

Let $(X, \mathcal{B}_X)$ be a measurable space and let $\mathbb{R}$ denote the real numbers with the Borel σ-algebra. Let $\mathcal{P}_1$ and $\mathcal{P}_2$ denote two classes of conditional probability measures on $\mathbb{R}$ given $x \in X$. The two arms will be characterized by conditional distributions $P_1(y|x) \in \mathcal{P}_1$ and $P_2(y|x) \in \mathcal{P}_2$, where we will assume throughout that for each x, every $P_i(y|x)$ has finite variance that is uniformly bounded over all $x \in X$. The actual conditional distributions are unknown but the classes $\mathcal{P}_1$ and $\mathcal{P}_2$ are known. Let $\mu_i(x)$ denote the regression function of $P_i(y|x)$. That is,

$$\mu_i(x) = E_{P_i}[y|x]$$

denotes the conditional mean of y given x under $P_i(\cdot|\cdot)$. Let $\mathcal{F}_1$ and $\mathcal{F}_2$ denote the two classes of regression functions generated by the distributions in $\mathcal{P}_1$ and $\mathcal{P}_2$, respectively, so that $\mathcal{F}_1, \mathcal{F}_2 \subset \mathcal{F}_X$ where $\mathcal{F}_X$ denotes the set of all measurable real-valued functions on X.

We obtain sequentially a set of points $x_1, x_2, \ldots \in X$ drawn i.i.d. according to some distribution P_X on X. At time n, based on x_n and our past observations, we select either arm 1 or arm 2. Let $\phi(x_n)$ denote our selection (ϕ also depends on the previous x_j, y_j, but we use the notation $\phi(x_n)$ to emphasize the dependence of the decision at time n on the observation x_n). We then obtain a reward y_n which is drawn from $P_{\phi(x_n)}(y|x_n)$ independent of all previous x_j and y_j. The cumulative reward $S_n = \sum_{j=1}^{n} y_j$ denotes the sum of our observations up to and including time n. Our goal is to maximize ES_n, the expected value of S_n.

If the distributions $P_1(\cdot|\cdot)$ and $P_2(\cdot|\cdot)$ (or actually just the regression functions $\mu_1(x)$ and $\mu_2(x)$) are known then the optimal policy would simply be to select the arm with the largest conditional mean given the observation x_n. Namely, the optimal policy would be to choose arm 1 if $\mu_1(x_n) \geq \mu_2(x_n)$ and choose arm 2 otherwise. Actually in the case $\mu_1(x_n) = \mu_2(x_n)$ the arms could be chosen arbitrarily. Thus, since the x_n are drawn i.i.d. according to some distribution P_X, the the optimal policy gives a cumulative payoff

$$ES_n^* = nE_{P_X} \max\{\mu_1(x), \mu_2(x)\} \tag{3}$$

The regret R_n is then defined as

$$R_n = nE_{P_X} \max\{\mu_1(x), \mu_2(x)\} - ES_n \tag{4}$$

Hence, our goal can be equivalently stated as trying to minimize the regret R_n. As in the standard bandit problem, the rate of growth of R_n is a measure of the performance of an algorithm ϕ, with a slower rate of growth indicating better performance. In the next two sections we provide some upper bounds on the regret under various conditions.

3 A General Upper Bound in Terms of Approximating the Regression Function

In this section, we provide a general upper bound on the regret for the bandit problem in terms of the sample size needed to approximate the regression function of each arm.

First, we formally define a notion of approximating the regression function of a conditional distribution from i.i.d. samples. Let $(X, \mathcal{B}_X)$ be a measurable space. Let $\mathcal{P}$ be a class of conditional probability measures on $\mathbb{R}$ given $x \in X$, and let P_X be a distribution on X. For each $P \in \mathcal{P}$, let $P_X P$ denote the joint distribution on $X \times \mathbb{R}$ given by $P_X(x)P(y|x)$ and let $\mathcal{D}_m(P_X P) = \{(x_1, y_1), \ldots, (x_m, y_m)\}$ denote m i.i.d. samples drawn according to $P_X P$, and let let $\mu(x) = E_P[y|x]$ denote the regression function of P. Finally, let $\mathcal{F}_X$ and let $\mathcal{F}_X$ denote the set of all measurable real-valued functions on X.

Definition 1 (Learning Regression Functions) *The class $\mathcal{P}$ will be said to be* learnable in absolute error *(or in $L^1(P_X)$) with sample size $m = m(\epsilon, \delta)$ if there exist measurable (with respect to $\mathcal{B}_X^m \otimes \mathcal{B}^m$, where $\mathcal{B}$ is the Borel σ-algebra on $\mathbb{R}^m$) mappings $A^m : (X^m \times \mathbb{R}^m) \to \mathcal{F}_X$ such that for all $\epsilon, \delta > 0$ and for every $P \in \mathcal{P}$,*

$$(P_X P)^m\{(x_1, \ldots, x_m, y_1, \ldots, y_m) : E_{P_X}|A^m(\mathcal{D}_m(P_X P)) - \mu(x)| > \epsilon\} < \delta$$

That is, there are mappings A^m such that with probability $1 - \delta$, after seeing m i.i.d. samples drawn according to P, A^m produces an estimate $\hat{\mu}(x) = A^m(\mathcal{D}_m(P_X P))$ that is ϵ-close to $\mu(x)$ in absolute error with respect to P_X. Learnability in squared error *(or in $L^2(P_X)$) can be defined similarly by requiring $[E_{P_X}(\hat{\mu}(x) - \mu(x))^2]^{1/2} < \epsilon$.*

This definition basically requires that we able to estimate the regression function of any $P \in P$ in probability using a sample size $m = m(\epsilon, \delta)$. The estimation criterion is essentially the same as that considered by Vapnik [16] and others in the statistics community and more recently in the computer science community (e.g., see Haussler [8]) under the name of PAC learning. The only difference between Definition 1 and the standard definition of PAC learning is that we have not explicitly introduced a hypothesis class. In fact, in the next section we will apply some results on PAC learning to provide bounds on the regret in terms of some structural properties of the class $\mathcal{P}$.

Our goal in this section is to obtain a bound on the regret for the bandit problem assuming that the classes $\mathcal{P}_1$ and $\mathcal{P}_2$ are learnable in the sense of Definition 1. The following lemma will be useful towards this end. This lemma shows that if the regression functions $\mu_1(x)$ and $\mu_2(x)$ for the two arms are approximately known, then the decision rule which selects the arms based on the approximate regression functions will be approximately optimal. Thus, this lemma provides a link between the bandit problem in which we wish to minimize regret and problem of learning (or estimating) the unknown regression functions.

Lemma 1 *If $E_{P_X}|\mu_1(x) - \hat{\mu}_1(x)| < \epsilon$ and $E_{P_X}|\mu_2(x) - \hat{\mu}_2(x)| < \epsilon$ then the expected contribution to the regret at each time using using a decision rule based on* $\max\{\hat{\mu}_1(x), \hat{\mu}_2(x)\}$ *is less than 2ϵ.*

Proof: Let $A = \{x : \mu_1(x) > \mu_2(x), \quad \hat{\mu}_1(x) < \hat{\mu}_2(x)\}$ and $B = \{x : \mu_1(x) < \mu_2(x), \quad \hat{\mu}_1(x) > \hat{\mu}_2(x)\}$

Note that if $x \notin A \cup B$ then the arm selected by choosing the larger of $\hat{\mu}_1(x)$ and $\hat{\mu}_2(x)$ is the optimal choice. In this case, the expected contribution to the regret is zero. On the other hand, if $x \in A \cup B$ then by selecting the larger of $\hat{\mu}_1(x)$, $\hat{\mu}_2(x)$, the algorithm will actually select the arm corresponding to the smaller of $\mu_1(x)$, $\mu_2(x)$. Thus, the expected contribution to the regret is

$$E_{P_X}[|\mu_1(x) - \mu_2(x)| I_{A \cup B}(x)] \tag{5}$$

where $I_{A \cup B}(x)$ is the indicator function of $A \cup B$.

But, for $x \in A$, we have $0 < \mu_1(x) - \mu_2(x) < \mu_1(x) - \hat{\mu}_1(x) + \hat{\mu}_2(x) - \mu_2(x)$ so that $|\mu_1(x) - \mu_2(x)| < |\mu_1(x) - \hat{\mu}_1(x)| + |\hat{\mu}_2(x) - \mu_2(x)|$. Similarly, for $x \in B$ we have $0 < \mu_2(x) - \mu_1(x) < \mu_2(x) - \hat{\mu}_2(x) + \hat{\mu}_1(x) - \mu_1(x)$ so that $|\mu_1(x) - \mu_2(x)| < |\mu_2(x) - \hat{\mu}_2(x)| + |\hat{\mu}_1(x) - \mu_1(x)|$. Therefore,

$$\begin{aligned} E_{P_X}[|\mu_1(x) - \mu_2(x)| I_{A\cup B}(x)] &\leq E_{P_X}\left[(|\mu_1(x) - \hat{\mu}_1(x)| + |\hat{\mu}_2(x) - \mu_2(x)|)\, I_{A\cup B}(x)\right] \\ &\leq E_{P_X}|\mu_1(x) - \hat{\mu}_1(x)| + E_{P_X}|\hat{\mu}_2(x) - \mu_2(x)| \\ &< 2\epsilon \end{aligned}$$

□

With this result we can construct a natural algorithm for the bandit problem assuming that the two classes $\mathcal{P}_1$ and $\mathcal{P}_2$ are learnable in the sense of Definition 1. The idea is to have learning intervals followed by selection/control intervals. In the learning interval L_j we will select arm 1 for some period followed by arm 2 for some period independent of the observations x_n. L_j will be chosen long enough so that we can learn $\mu_1(x)$ and $\mu_2(x)$ to some prescribed accuracy and confidence (ϵ_j, δ_j). In the control interval C_j, given an observation x_n we will choose an arm based on our estimates for $\mu_1(x)$ and $\mu_2(x)$. That is, we will choose the arm corresponding to the larger of $\hat{\mu}_1(x_n)$ and $\hat{\mu}_2(x_n)$. Thus, in the learning intervals we may suffer by ignoring the observations x_n, but we gain knowledge about the unknown distributions which is exploited in the control intervals in order to minimize the regret.

Theorem 1 *Let $\mathcal{P}_1$ and $\mathcal{P}_2$ be two classes of conditional distributions on $\mathbb{R}$ given $x \in X$, and let $\mathcal{F}_1$ and $\mathcal{F}_2$ be the classes regression functions of the distributions in $\mathcal{P}_1$ and $\mathcal{P}_2$, respectively. Suppose for some $M < \infty$, $0 \leq f(x) \leq M$ for all $f \in \mathcal{F}_1 \cup \mathcal{F}_2$ and all $x \in X$. Let P_X be a probability measure on X. For $i = 1, 2$, suppose $\mathcal{P}_i$ is learnable in absolute error (Definition 1) to accuracy ϵ and confidence $1 - \delta$ using $m_i(\epsilon, \delta)$ samples. Let*

$$m(\epsilon) = m_1(\epsilon, \epsilon/2) + m_2(\epsilon, \epsilon/2) \quad \text{and} \quad \ell(\epsilon) = \frac{1}{\epsilon} m(\epsilon)$$

and suppose that $m(\epsilon)$ satisfies

$$m'(\epsilon) < -c\frac{m(\epsilon)}{\epsilon}$$

for some $c > 0$. Then given any $P_1(y|x) \in \mathcal{P}_1$ and $P_2(y|x) \in \mathcal{P}_2$ there exists an algorithm such that the expected regret R_n satisfies $R_n < (4/\ln 2)(1 + 1/c)(M + 1)\, n\, \ell^{-1}(n)$.

Proof: Choose ϵ_j so that $\ell(\epsilon_j) = \frac{1}{\epsilon_j} m(\epsilon_j) = 2^j$. We will let the length of the learning interval L_j be $m(\epsilon_j)$ and the length of the control interval C_j be $\ell(\epsilon_j)$, so that the length of the entire stage j (L_j together with C_j) is $m(\epsilon_j) + \frac{1}{\epsilon_j} m(\epsilon_j)$. Our decision rule is as follows. In the learning interval L_j we select arm 1 for the first $m_1(\epsilon_j, \epsilon_j/2)$ samples and we select arm 2 for the next $m_2(\epsilon_j, \epsilon_j/2)$ samples regardless of the observations x_n. Using the observations from the learning interval L_j we form estimates $\hat{\mu}_1(x)$ and $\hat{\mu}_2(x)$ for the regression functions $\mu_1(x)$ and $\mu_2(x)$ respectively. Note that by our choice of the length of the interval L_j with probability greater than $1 - \epsilon_j$ both regression function estimates will simultaneously be ϵ_j-close to the corresponding true regression functions. Then in the control interval C_j, after observing x_n we select an arm according to the larger of $\hat{\mu}_1(x_n)$ and $\hat{\mu}_2(x_n)$.

Now, since the expected regret at each time is certainly bounded by M, the expected regret for the entire learning interval L_j is bounded by $Mm(\epsilon_j)$. For the control interval C_j, as previously mentioned, with probability $1 - \epsilon_j$ both $\mu_1(x)$ and $\mu_2(x)$ will be learned to accuracy ϵ_j. In this case, by Lemma 1, the expected regret at each time in C_j is less than $2\epsilon_j$ so that the expected regret for the whole interval C_j is less than $2\epsilon_j(\frac{1}{\epsilon_j}m(\epsilon_j)) = 2m(\epsilon_j)$. In the case that we may fail to learn either $\mu_1(x)$ or $\mu_2(x)$ to accuracy ϵ_j, the regret at each time is still bounded by M, so that the regret for the whole interval C_j is bounded by $\frac{M}{\epsilon_j}m(\epsilon_j)$. However, this case happens with probability less than ϵ_j. Thus, the expected regret for the interval C_j is bounded by $(1-\epsilon_j)2m(\epsilon_j) + \epsilon_j(Mm(\epsilon_j)/\epsilon_j) < (M+2)m(\epsilon_j)$. Thus, the expected regret for the entire stage j (L_j and C_j) is less than $2(M+1)m(\epsilon_j)$.

For any n, let $k(n)$ be a positive integer such that

$$n \le \ell(\epsilon_{k(n)}) = \frac{m(\epsilon_{k(n)})}{\epsilon_{k(n)}} < 2n \tag{6}$$

Since the length of the control interval $C_{k(n)} = \ell(\epsilon_{k(n)})$ is itself greater than n, the number of stages (L_j together with C_j) which are encountered by time n (including the the stage that may be partially underway at time n) is certainly less than $k(n)$. Therefore, the expected regret R_n up to and including time n is bounded by the sum of the expected regret over each of the first $k(n)$ stages. Namely,

$$R_n \le 2(M+1)\sum_{j=1}^{k(n)} m(\epsilon_j) \tag{7}$$

Now, we show that the last term of the sum in equation (7) dominates. For each x, let $\epsilon(x)$ be such that

$$\frac{m(\epsilon(x))}{\epsilon(x)} = 2^x$$

and let $g(x) = m(\epsilon(x))$. Then $g(x) = 2^x\epsilon(x)$ so that

$$g'(x) = (\ln 2)2^x\epsilon(x) + 2^x\epsilon'(x) = g(x)\left[\ln 2 + \frac{\epsilon'(x)}{\epsilon(x)}\right] \tag{8}$$

Also, $m'(\epsilon(x))\epsilon'(x) = (\ln 2)2^x\epsilon(x) + 2^x\epsilon'(x)$ so that

$$\frac{\epsilon'(x)}{\epsilon(x)} = \frac{\ln 2}{m'(\epsilon(x))2^{-x} - 1} > \frac{\ln 2}{(-cm(\epsilon(x))/\epsilon(x))2^{-x} - 1} = \frac{-\ln 2}{1+c} \tag{9}$$

where we have used the condition on $m'(\epsilon)$ in the statement of the theorem. From equations (8) and (9) we have

$$g'(x) > \frac{c\ln 2}{1+c}g(x)$$

Hence,

$$\begin{aligned}
\sum_{j=1}^{k(n)} m(\epsilon_j) &\le \int_0^{k(n)} m(\epsilon(x))\,dx \\
&< \frac{1+c}{c\ln 2}\int_0^{k(n)} g'(x)\,dx \\
&\le (1/\ln 2)(1+1/c)g(k(n)) \\
&= (1/\ln 2)(1+1/c)m(\epsilon_{k(n)})
\end{aligned} \tag{10}$$

From equations (7) and (10) we have

$$R_n < (2/\ln 2)(1+1/c)(M+1)m(\epsilon_{k(n)}) = (2/\ln 2)(1+1/c)(M+1)\epsilon_{k(n)}[m(\epsilon_{k(n)})/\epsilon_{k(n)}]$$

Finally, from the choice of $k(n)$ (equation (6)) and the fact that $\ell(\epsilon)$ is decreasing in ϵ, we have $\epsilon_{k(n)} < \ell^{-1}(n)$ and $m(\epsilon_{k(n)})/\epsilon_{k(n)} < 2n$, from which the result follows. □

4 Bounds on Regret Via PAC Learning Bounds

In this section, we use some results on PAC learning in conjunction with Theorem 1 to obtain upper bounds on the regret in terms of some structural properties of the classes $\mathcal{P}_1$ and $\mathcal{P}_2$ (actually in terms of the classes $\mathcal{F}_1$ and $\mathcal{F}_2$ of regression functions).

PAC learning results provide sample size bounds for non-parametric estimation from random samples in terms of certain "complexity" parameters (namely covering numbers and VC/pseudo dimension) of the hypothesis class used in the estimation. Many early and fundamental results on PAC-like models have been obtained in the statistics community by Vapnik and Chervonenkis [16, 15], Dudley [6], Pollard [10], and others. The work of Valiant [14] generated interest in similar problems in the computer science community (in fact, the term "PAC learning" arose from this later work), and the recent paper by Haussler [8] provides a general framework that consolidates much of the previous work on the PAC model.

Here we give only a brief description of two results on PAC learning stated in a form suited to the present paper. We refer the reader to [16, 8] for more details, further results, and additional references.

Let $\mathcal{P}$ be a collection of conditional probability measures on $\mathbb{R}$ given $x \in X$ as before, and let P_X be a probability measure on X. In Definition 1, if the range of the mapping is some function class $\mathcal{F}$, PAC learning results provide conditions on $\mathcal{F}$ and sample size bounds sufficient for learning. We will consider only the case in which $\mathcal{F}$ denotes the set of real-valued regression functions on X induced by the conditional distributions in $\mathcal{P}$.

Note that since we wish to approximate the regression function $\mu(x)$ our learning criterion will be chosen as the squared loss. Namely, we will evaluate the quality of a function $f(x)$ on data $(x_1, y_1), \ldots, (x_m, y_m)$ using the criterion $\left(\sum_{j=1}^{m}(f(x_j) - y_j)^2\right)^{1/2}$ and the quality of $f(x)$ with respect to the distribution $P_X P = P_X(x)P(y|x)$ generating the data as $[E_{P_X P}(f(x) - y)^2]^{1/2}$. With this criterion, the regression function $\mu(x)$ will be the optimal rule.

Regarding the distribution P_X on X, two extreme cases are most commonly studied. One extreme is the case where P_X fixed and known. The other extreme case is the distribution-free case in which learning is required to take place for every distribution P_X on X, and moreover there must exist a sample size which works uniformly for all distributions. One motivation for this requirement is that in certain situations one may not know and may not be able to control the distribution P_X.

In the case of a fixed and known distribution P_X on X, the notion of covering number/metric entropy can be used to provide a sufficient condition for learnability.

Definition 2 (Covering Number and Metric Entropy) *Let (Y, ρ) be a metric space. Define $N(\epsilon) \equiv N(\epsilon, Y, \rho)$ to be the smallest integer n such that there exists $y_1, \ldots, y_n \in Y$ with $Y = \cup_{j=1}^{n} B_\epsilon(y_j)$ where $B_\epsilon(y_j)$ is the open ball of radius ϵ centered at y_j. If no such n exists, then $N(\epsilon, Y, \rho) = \infty$. The set $\{y_1, \ldots, y_n\}$ is said to form an* ϵ-cover *of Y, $N(\epsilon)$ is called the* covering number *of Y and $\log_2 N(\epsilon)$ is called the* metric entropy *of Y*

These notions are equally valid in the case where ρ is only a pseudo-metric rather than a metric (i.e., even if $\rho(y_1, y_2) = 0$ does not imply $y_1 = y_2$). For the learning problem under consideration, the measure of error $d_{P_X}(f_1, f_2) \equiv [E_{P_X}(f_1(x) - f_2(x))^2]^{1/2}$ between two functions f_1, f_2 with respect to a distribution P_X is a pseudo-metric. Note that by the Cauchy-Schwarz inequality,

$$E_{P_X}|f_1(x) - f_2(x)| \leq [E_{P_X}(f_1(x) - f_2(x))^2]^{1/2} = d_{P_X}(f_1, f_2) \tag{11}$$

Hence, if we can form an estimate $\hat{\mu}(x)$ of the regression function such that $d_{P_X}(\hat{\mu}, \mu) < \epsilon$ then we will also have $E_{P_X}|\hat{\mu}(x) - \mu(x)| < \epsilon$ so that we will be able to apply Lemma 1 in the context of the bandit problem.

For the case of a fixed distribution, one can obtain the following result (e.g., see Vapnik [16], Haussler [8]).

Theorem 2 *Let $\mathcal{P}$ be a collection of conditional distributions on $\mathbb{R}$ given $x \in X$. Let $\mathcal{F}$ denote the set of regression functions induced by the distributions in $\mathcal{P}$ and suppose that $0 \leq f(x) \leq M$ for all $f \in \mathcal{F}$ and all $x \in X$. Let P_X be a fixed distribution on X, and suppose that $N(\epsilon, \mathcal{F}, d_{P_X}) < \infty$ for all $\epsilon > 0$. Then $\mathcal{P}$ is learnable with respect to squared error loss (in the sense of Definition 1) using a sample size*

$$m(\epsilon, \delta) = \frac{CM^2}{\epsilon^2} \ln \frac{2N(\epsilon/2, \mathcal{F}, d_{P_X})}{\delta}$$

for some constant C.

Note that in the above result, it is actually not necessary for every $f \in \mathcal{F}$ to be bounded. We really only need the functions in the ϵ-cover to be bounded, and the bound can depend on ϵ. Specifically, in the sample size bound in the above theorem M could be replaced by $M(\epsilon/2)$ where $M(\epsilon/2)$ is a bound on all the functions in an $\epsilon/2$-cover.

In the case of distribution-free learning, a quantity known as the VC dimension (or pseudo dimension) [16, 8] can be used to provide a sufficient condition for learnability.

Definition 3 (Pseudo Dimension) *Let $\mathcal{F}$ be a collection of functions from a set X to $\mathbb{R}$. For any set of points $\overline{x} = (x_1, \ldots, x_d)$ from X, let $\mathcal{F}_{|\overline{x}} = \{(f(x_1), \ldots, f(x_d)) : f \in \mathcal{F}\}$. $\mathcal{F}_{|\overline{x}}$ is a set of points in $\mathbb{R}^d$. If there is some translation of $F_{|\overline{x}}$ which intersects all of the 2^d orthants of $\mathbb{R}^d$ then $\overline{x}$ is said to be* shattered *by $\mathcal{F}$. The* pseudo dimension *of $\mathcal{F}$, which we denote* $\dim(\mathcal{F})$, *is the largest integer d such that there exists a set of d points in X that is shattered by $\mathcal{F}$. If no such largest integer exists then* $\dim(\mathcal{F})$ *is infinite.*

The following theorem (e.g., see [16] or [8]) gives a sufficient condition and sample size bounds for distribution-free learnability. Certain measurability conditions are actually required for the functions (see [16, 8]), but we will not concern ourselves these conditions here and will assume that $\mathcal{F}$ satisfies the required conditions.

Theorem 3 *Let $\mathcal{P}$ be a collection of conditional distributions on $\mathbb{R}$ given $x \in X$. Let $\mathcal{F}$ denote the set of regression functions induced by the distributions in $\mathcal{P}$ and suppose that $0 \leq f(x) \leq M$ for all $f \in \mathcal{F}$ and all $x \in X$. Suppose that* $\dim(\mathcal{F}) = d < \infty$. *Then $\mathcal{P}$ is learnable for every distribution P_X on X (and P_X need not be known beforehand) using a sample size*

$$m(\epsilon, \delta) = \frac{C_1 M^2}{\epsilon^4} \left(2d \ln \frac{C_2 M}{\epsilon^2} + \ln \frac{8}{\delta}\right)$$

for some constant C.

Theorems 2 and 3 can be used in conjunction with Theorem 1 to give bounds on the regret for general classes of conditional densities $\mathcal{P}_1$ and $\mathcal{P}_2$ for which the corresponding classes of regression functions have finite covering numbers or finite pseudo dimension.

Corollary 1 *Let $\mathcal{P}_1$ and $\mathcal{P}_2$ be classes of conditional densities and let $\mathcal{F}_1$ and $\mathcal{F}_2$ be the corresponding classes of regression functions. Suppose that $0 \leq f(x) \leq M$ for every $f \in \mathcal{F}_1 \cup F_2$ and every $x \in X$. Let P_X be a fixed distribution on X.*

a) *If* $N(\epsilon, \mathcal{F}_i, d_{P_X}) = o(1/\epsilon^d)$ *for some* $d < \infty$ *and* $i = 1, 2$ *then there exists an algorithm such that* $R_n = o(n^{2/3+a})$ *for every* $a > 0$.

b) *If* $N(\epsilon, \mathcal{F}_i, d_{P_X}) = o(\exp(1/\epsilon^d))$ *for some* $d < \infty$ *and* $i = 1, 2$ *then there exists an algorithm such that* $R_n = o(n^{(d+2)/(d+3)+a})$ *for every* $a > 0$.

c) *If* $N(\epsilon, \mathcal{F}_i, d_{P_X}) = o(\exp[\exp(1/\epsilon)])$ *for* $i = 1, 2$ *then there exists an algorithm such that* $R_n = o(n/(\ln n)^{1-a})$ *for every* $a > 0$.

Corollary 2 *Let* $\mathcal{P}_1$ *and* $\mathcal{P}_2$ *be classes of conditional densities and let* $\mathcal{F}_1$ *and* $\mathcal{F}_2$ *be the corresponding classes of regression functions. Suppose that* $0 \leq f(x) \leq M$ *for every* $f \in \mathcal{F}_1 \cup F_2$ *and every* $x \in X$. *If* $dim(\mathcal{F}_i) < d$ *for some* $d < \infty$ *and* $i = 1, 2$ *then there exists an algorithm such that for every distribution* P_X *on* X $R_n = o(n^{4/5+a})$ *for every* $a > 0$.

Note that the standard two-armed bandit problem is a special case of our formulation in which for every $P_i \in \mathcal{P}_i$, $P_i(y|x)$ does not depend on x, or alternatively that $P_X = \delta_{x_0}$. In these cases, $N(\epsilon, \mathcal{F}_i, d_{P_X}) = O(1/\epsilon)$ so that from part (a) of Corollary 1 we get $R_n = o(n^{2/3+a})$ for every $a > 0$. Although this bound is significantly worse than the $\log n$ bound of [9], our bound holds for much larger classes of conditional probabilities as well. Furthermore, we do not require the regularity conditions in [9]. Finally, we believe our approach and bounds can be improved as briefly described in the next section, which may allow us to get a $\log n$ bound for the special case of the standard two-armed bandit.

5 Conclusions

We introduced a general class of bandit-type problems in which we must select one of two arms based on the observation of an auxiliary variable, with the goal of maximizing the expected cumulative reward. We provided upper bounds on the regret for such problems in terms of the sample size required to approximate the regression functions of the two arms. In conjunction with some results from PAC learning, our results provide bounds on the regret for wide range of classes of conditional distributions in terms of certain complexity parameters (covering numbers, pseudo dimension) of the class of regression functions. The paper provides preliminary results towards what we believe is an interesting class of problems, and also establishes a connection between bandit problems and work in non-parametric regression estimation.

There are a number of directions that may be interesting to pursue further. First, we believe the upper bounds provided can be improved in various ways. For example, it is not necessary to estimate both regression functions to arbitrarily small accuracy over the entire domain. At each point, the regression functions need only be estimated to an accuracy ϵ sufficient to resolve the larger of the regression functions. Of course, we still need to let the confidence parameter δ become arbitrarily small, but the dependence of the sample size on δ is much weaker than the dependence on ϵ. Taking this into account could significantly improve the bounds on the regret. Also, our bounds hold for quite general classes. One would certainly expect that tighter bounds could be obtained for certain specific classes. An interesting direction to consider is to obtain lower bounds on the regret. In contrast to the classical bandit problems, there are two factors which should contribute to a lower bound on the achievable regret — the usual uncertainty due to the randomness in the rewards as well as the complexity of learning the underlying regression function. It may be necessary to impose certain regularity conditions in order to get useful lower bounds. Another interesting direction may be to consider bandit problems under other sampling schemes on the auxiliary variable x. That is, one could consider worst-case settings in which the x_n are chosen by an adversary, or mixed settings as in [11]. Finally, many variations of the classical bandit problem (such as multi-armed bandits, discounted rewards, Bayesian settings, etc.) could also be considered in the setting of this paper.

Acknowledgments

I would like to thank O. Zeitouni for helpful discussions.

References

[1] R. Agrawal, M. Hedge, D. Teneketzis, "Asymptotically efficient adaptive allocation rules for the multiarmed bandit problem with switching cost," *IEEE Trans. Automatic Control*, Vol. AC-33, No. 10, pp. 899-906, 1988.

[2] R. Agrawal, D. Teneketzis, V. Anantharam, "Asymptotically efficient adaptive allocation schemes for controlled i.i.d. processes: finite parameter space," *IEEE Trans. Automatic Control*, Vol. AC-34, No. 3, pp. 258-266, 1989.

[3] V. Anantharam, P. Varaiya, J. Walrand, "Asymptotically efficient allocation rules for the multiarmed bandit problem with multiple plays — part I: i.i.d. rewards," *IEEE Trans. Automatic Control*, Vol. AC-32, No. 11, pp. 968-976, 1987.

[4] V. Anantharam, P. Varaiya, J. Walrand, "Asymptotically efficient allocation rules for the multiarmed bandit problem with multiple plays — part II: Markovian rewards," *IEEE Trans. Automatic Control*, Vol. AC-32, No. 11, pp. 977-982, 1987.

[5] D.A. Berry and B. Fristedt, *Bandit Problems*, Chapman and Hall, 1985.

[6] R.M. Dudley, "Central limit theorems for empirical measures," *Ann. Probability*, Vol. 6, No. 6, pp. 899-929, 1978.

[7] J.C. Gittins, *Multi-armed Bandit Allocation Indices*, Wiley & Sons, 1989.

[8] D. Haussler, "Decision theoretic generalizations of the PAC model for neural net and other learning applications," *Information and Computation*, Vol. 100, pp. 78-150, 1992.

[9] T.L. Lai and H. Robbins, "Asymptotically efficient adaptive allocation rules," *Advances in Applied Mathematics*, Vol. 6, pp. 4-22, 1985.

[10] D. Pollard, *Convergence of Stochastic Processes*, Springer-Verlag, 1984.

[11] S.E. Posner and S.R. Kulkarni, "On-line learning of functions of bounded variation under various sampling schemes," Proc. of Sixth Annual Conference on Computational Learning Theory, pp. 439-445, 1993.

[12] H. Robbins, "Some aspects of the sequential design of experiments," *Bull. Amer. Math. Soc.*, Vol. 58, pp. 527-535, 1952.

[13] W.R. Thompson, "On the likelihood that one unknown probability exceeds another in view of the evidence of two samples," *Biometrika*, Vol. 25, pp. 275-294, 1933.

[14] L.G. Valiant, "A theory of the learnable," *Comm. ACM*, Vol. 27, No. 11, pp. 1134-1142, 1984.

[15] V.N. Vapnik and A.Ya. Chervonenkis, "On the uniform convergence of relative frequencies to their probabilities," *Theory of Prob. and its Appl.*, Vol. 16, No. 2, pp. 264-280, 1971.

[16] V.N. Vapnik, *Estimation of Dependences Based on Empirical Data*, Springer-Verlag, 1982.

[17] M. Woodroofe, "A one-armed bandit problem with a concomitant variable," *Journal of the American Statistical Association*, Vol. 74, No. 368, pp. 799-806, 1979.

Formation and Agreement Problems for Anonymous Mobile Robots*

Ichiro Suzuki † Masafumi Yamashita ‡
† Department of Electrical Engineering and Computer Science
University of Wisconsin–Milwaukee, Milwaukee, WI 53201
‡ Department of Electrical Engineering, Faculty of Engineering
Hiroshima University, Higashi-Hiroshima 724, Japan
Email: suzuki@cs.uwm.edu, mak@se.hiroshima-u.ac.jp

Abstract A system consisting of multiple mobile robots in which the robots can see each other by their eye sensors but are not equipped with any communication system, can be viewed as a distributed system in which the components (i.e., robots) can "communicate" with each other only by means of their moves. We use this system to investigate, through a case study of a number of problems on the formation of geometric figures in the plane, the power and limitations of the distributed control method for mobile robots. In the distributed control method, at every tick of its local clock, each robot observes the positions of all the robots and moves to a new position determined by the given algorithm. The robots are anonymous in the sense that they all execute the same algorithm and they cannot be distinguished by appearances. The local clocks of the robots are not necessarily synchronized, and initially, the robots do not have a common x-y coordinate system. The problems we discuss include (1) converging the robots to a single point, (2) moving the robots to a single point, (3) agreement on a single point, (4) agreement on the unit distance, (5) agreement on direction, and (6) leader election. We develop algorithms for solving some of these problems under various conditions. Some impossibility results are also presented.

1 Introduction

In the last several years, interest in the distributed control method for multiple mobile robots has increased considerably [1, 2, 5, 7]. The main idea of the method is to let each robot execute a simple algorithm and determine its movement adaptively based on the observed movement of other robots, so that the robots as a whole group will achieve the given goal. This approach has been shown to be very promising for the generation of certain patterns and collision avoidance. In the earlier works on distributed robot control, the main emphasis is on the development of heuristic algorithms for various problems and the presentation of simulation results, and in many cases, formal discussions on the correctness and performance of the algorithms are not given [1] [5].

A robot system in which the robots can communicate with each other by radio, such as a system of radio-controlled vehicles or spaceships, can be considered as a distributed system whose communication topology is a complete graph. Therefore, such systems can be analyzed using the standard techniques developed for distributed computing systems (although such analyses are by no means easy). In this paper, we consider a system consisting of multiple

---

*This work was supported in part by the National Science Foundation under grants CCR-9004346 and IRI-9307506.

mobile robots in which the robots can see each other by their eye sensors, but they are not equipped with any communication system. We view this system as a distributed system in which the components (i.e., robots) can "communicate" with each other only by means of their moves. Clearly this system is a suitable model for studying the distributed control method mentioned in the previous paragraph. As the reader might expect from the assumptions we make about the robots below, the study reveals delicate interplay of a number of key concepts of distributed computing, such as synchrony and asynchrony, communication, termination detection, self-stabilization, anonymity of processors, and knowledge (in a casual sense).

A basic problem for such a robot system is to design an algorithm such that, if all the robots execute it individually, then the robots as a whole group will eventually form the given geometric figure, such as a circle and a line segment [3, 5, 6]. The main goal of this paper is to present some theoretical results related to this problem. The results presented here provide useful insights that will help us to answer certain fundamental questions, such as whether the given algorithm really solves the given problem and, for that matter, whether the given problem is solvable at all in a strict sense by a distributed algorithm. This work is a step toward the ultimate goal of determining exactly what class of problems are solvable in a distributed manner.

We assume that each robot is a mobile processor having an eye sensor which, at every tick of its local clock, observes the positions of all the robots (including itself) in terms of its own local x-y coordinate system, and moves to a new position determined by the given algorithm. To simplify the discussion, in this paper we assume that a robot can move to its new position instantaneously. This assumption helps us to bring out the fundamental issues of the problem, and still, many of the techniques and results we obtain for this simplified case seem to apply (with some modifications) to the case when the move of a robot is not instantaneous. (We plan to report on the noninstantaneous case in a future paper.)

The robot (or the algorithm it uses to compute the new position) is *oblivious* if the new position is determined only from the positions of the robots observed at that time instant. Otherwise, it is not oblivious, and the new position may depend also on the observations made in the past. The local clocks of the robots are not necessarily synchronized, i.e., they may tick at different rates. It is assumed that initially, the robots do not have a common x-y coordinate system. So the local x-y coordinate systems of two robot may not agree on any of the following: the location of the origin, the unit distance, and the direction of the positive x-axis. Finally, we assume that the robots are anonymous, in the sense that they all use the same algorithm for determining the next position, and they cannot be distinguished by their appearances. This implies, for example, that depending on what movements are allowed, a robot that observes other robots at two consecutive clock ticks may not be able to tell which robot has moved to which position between the two ticks.

Under these assumptions, we first consider the problem of converging all the robots toward a single point. (The process of convergence need not terminate in finite steps.) Note that since the robots do not have a common x-y coordinate system, we cannot simply use an algorithm such as "move toward point $(0, 0)$". For this problem, we give an oblivious algorithm and then discuss the subtlety of the problem by showing how certain minor changes in the algorithm or the assumption affect the possibility of achieving the goal. Also, we show that the set of geometric figures realizable by any deterministic algorithm must necessarily include the configuration in which all robots are located at the same position.

Next, we investigate the problem of having the robots agree on a common x-y coordinate system. Clearly, such an agreement can greatly reduce the complexity of subsequent motion coordination algorithms. The problem consists of three subproblems, agreement on the origin, agreement on the unit distance, and agreement on the direction of the positive x-axis. We first consider a related problem of moving two robots to a single point *in finite steps* (such a problem is called a *formation problem*, in contrast to a convergence problem) and show that this problem can be solved by a nonoblivious algorithm but not by any oblivious algorithm. (The corresponding convergence problem can be solved by an oblivious algorithm, as we stated

```
1.  $I := \emptyset$;
2.  repeat
3.      Wait until $c_r$ ticks; Let $k$ be the new value of $c_r$;
        /* Start Move Action */
4.      Let $P_r$ be the multiset of positions of the robots, excluding $r$, at local time $k$,
            given in terms of $Z_r$;
5.      $I := I \cup \{(p,k) | p \in P_r\} \cup \{(p^*,k)\}$, where $p^*$ is the current position of $r$;
6.      Randomly select an element $q$ from $\psi(I)$;
7.      Move to $q$;
        /* End Move Action */
8.  until forever.
```

Figure 1: Behavior of robot r.

above.) Since two robots can agree on the origin if they can move to a single point, this result shows that agreement on the origin for two robots is solvable by a nonoblivious algorithm. Moreover, we show that agreement on the origin is solvable by a nonoblivious algorithm even for the case $n > 2$, where n is the number of robots. We also show that agreement on the unit distance is solvable by a nonoblivious algorithm if either $n = 2$, or $n > 2$ and no two robots are "clones" (to be defined later) of each other. On the other hand, we show that agreement on direction is not solvable even if $n = 2$ and the clocks are synchronized. This last result shows that the robots cannot agree on a common x-y coordinate system in general.

Finally, we consider the case in which the robots have a sense of direction (i.e., the direction of the positive x-axis is the same for all robots). For this case, we show that the robots can agree on a common x-y coordinate system and elect a leader, provided that no "clones" exist. We can show that once a unique leader is elected, the robots can be moved to form any geometric figure.

It can be shown that if the local clocks are synchronized, then the robots can easily communicate with each other by means of the distances of their moves, once they agree on the unit distance. Therefore, many problems on synchronized robot systems can be reduced to the corresponding problems on anonymous complete networks, and can be analyzed using the techniques developed, for example, in [8]. Details can be found in the full paper.

We present necessary definitions and basic assumptions in Section 2. Section 3 discusses the problem of converging the robots to a point. Agreement on a common coordinate system is discussed in Section 4. In Section 5 we consider the case in which the robots have a sense of direction. Concluding remarks are found in Section 6. In this abstract, we present only the key ideas. All the details and the missing proofs can be found in the full paper.

2 Definitions and Basic Assumptions

We briefly formalize the problem described in Section 1.

A robot r is a mobile processor with sufficiently large memory space, an artificial eye sensor, a local x-y coordinate system Z_r, and a local clock c_r. The behavior of r is given in Figure 1. Local clock c_r is an integer counter whose contents, initially 0, continue to increase by 1 (i.e., the clock "ticks") at infinitely many (unknown) real time instants. We assume that at real time 0, r is at the origin $(0,0)$ of Z_r, executes line 1 instantaneously, and waits (in line 3) for its local clock c_r to tick and reach the value 1. The robots have no access to real time. The symbol I, called the *configuration* of the memory of r, is a multiset[1] containing elements of the form

---

[1] For simplicity, we assume that a robot is a point, and hence two or more robots can occupy the same position

1. (p, k), where p is the position of a robot other than r that r observed at local time k, and
2. $\langle p, k\rangle$, where p is the position of r at local time k.

Here, p is given in terms of Z_r. The function ψ (line 6) is called an *algorithm* for r, where for each configuration I, $\psi(I)$ is the set of possible next positions of r (given in terms of Z_r) when r "knows" I. If $\psi(I)$ is a singleton for all I, then ψ is said to be *deterministic.* Otherwise, r randomly selects its next position from $\psi(I)$ (line 6), and hence ψ is a *randomized* algorithm. The fact that I contains elements of the form $\langle p, k\rangle$ indicates that r always knows its current position in terms of Z_r. Also, note that I contains only the *positions* of the robots that r has observed. So r never observes, for example, the velocity of other robots. Note that the robots are anonymous in the following sense: (1) the initial configuration $\emptyset$ and function ψ are common to all the robots, (2) the identifier "r" of robot r is not an argument of ψ, and (3) P_r contains only the positions of the robots (but not their identities).

We assume that different robots may have different coordinate systems, i.e., $Z_r \neq Z_s$ for some robots r and s. Local clocks of two robots are said to be *synchronized* if they always tick simultaneously; otherwise, they are *asynchronous.* Unless otherwise stated, we assume that the local clocks are not necessarily synchronized. Therefore, even though all the robots execute line 1 at real time 0, they may not execute line 4 for the first time simultaneously. This means that the robots may not be able to obtain a consistent view of their initial distribution. In fact, we will see that the difficulty of obtaining a consistent view of the system is a source of some of the technical problems in designing correct algorithms. For example, if all the robots know that they can observe their initial distribution simultaneously, then they can adopt the center of gravity of their distribution as the common origin, and the minimum distance between any two robots as the common unit distance. So the robots can move to a point on the circumference of the unit circle centered at the origin, and form an approximation of a circle.

Robot r is said to be *active* when it executes lines 4–7; otherwise, it is *inactive.* By the definition of c_r, r becomes active infinitely many times. As we stated in Section 1, we assume that a robot can move to any position instantaneously, and thus the time it takes to execute lines 4–7 is negligibly small.

Using this framework, it is possible to discuss the situation in which some robots are added and/or removed from the system dynamically. This can be done by assuming that a robot becomes visible (or invisible) when it is added (or removed) from the system. Under this assumption, all the algorithms we present in this paper can easily be modified to work correctly even if the number of robots changes a finite number of times. (See the full paper for details.) Algorithms having this property can be viewed as self-stabilizing algorithms, since they solve the given problem in the presence of transient failures. This is an advantage of the distributed control method. In the centralized method, the entire system can crash if the robot controlling all other robots becomes faulty (and is removed).

Fix an x-y coordinate system Z. (The robots have no information on Z.) Let π be a predicate over the set of multisets of points (given in terms of Z) that is invariant under any motion (i.e., rotation and parallel transformation) and uniform scaling. For example, π might be true iff the given points are on the circumference of a circle or on a line segment. For such π, we consider two types of problems, the *convergence problem* and the *formation problem.* In the convergence problem, the goal is to design an algorithm ψ such that, if all the robots execute the instructions using ψ, then the positions of the robots converge to a multiset of points (with respect to the absolute coordinate system Z) satisfying π, regardless of the initial distribution of the robots. The formation problem is similar, except that the robots must reach some points satisfying π in finite steps. Since the robots have no knowledge of the coordinate system Z, all we can expect is to have the robots converge to or form a figure which is similar to the given

---

simultaneously. If robot r observes at local time k that both robots r_1 and r_2 are at position p, then I should contain (p, k) for each of r_1 and r_2.

goal figure. The restriction on π stated above was introduced for this reason. All predicates we discuss in the following are assumed to satisfy this condition.

3 Converging to a Point

Define a predicate π_{point} by $\pi_{point}(p_1, \ldots, p_n) = true$ iff $p_i = p_j$ for any $1 \leq i, j \leq n$, where $n \geq 2$ is the number of robots. In this section we discuss the convergence problem for π_{point}, which we call POINT. That is, POINT is the problem of converging all the robots to a single point. We present three correct oblivious algorithms and one incorrect algorithm. All of these are deterministic algorithms. We also discuss a limitation of deterministic algorithms.

Our first algorithm, ψ_{point}, simply moves each robot r to the midpoint of its current position and the position of a robot that is currently furthest from r. Formally, let p^* and P_r be the current position of r and the multiset of positions of all other robots at local time k, respectively, as in lines 3 and 4 of Figure 1. Since ψ_{point} is oblivious, "$\psi_{point}(I)$" can simply be written as "$\psi_{point}(P_r, p^*)$." Then $\psi_{point}(P_r, p^*) = \{(p^* + p)/2\}$, where p is any of the points in P_r furthest from p^*.

Theorem 1 *Function ψ_{point} correctly solves problem POINT.*

Outline of Proof For any real time t, let $S(t)$ be the set of positions of the robots at t. Then it suffices to show that $CH(S(t))$ converges to a single point as t goes to infinity, where CH denotes the convex hull. Suppose this is not the case. Since $CH(S(t)) \supseteq CH(S(t'))$ for any real times t and t' such that $t < t'$, $CH(S(t))$ must then converge to a convex polygon C. This implies that for any small real number $\delta > 0$, there exists some real time t_0 such that for any $t > t_0$ and any corner v of C, there is at least one robot in the δ-neighborhood of v (denoted δ_v). If δ is chosen sufficiently small, then any such inactive robot must eventually (become active and) leave δ_v, since the robots that are furthest from it are located near the δ-neighborhood of some other corners of C, that are more than 3δ away from v. So, for $CH(S(t))$ to converge to C, some of the robots, say r, must enter δ_v repeatedly. However, since δ is chosen sufficiently small, this is possible only if there already is a robot in δ_v when r chooses a robot furthest from itself. Clearly this is impossible. □

Let us modify ψ_{point}, and use point p (the point in P_r furthest from p^*) instead of $(p^* + p)/2$ as the next position of r. That is, robot r moves to the position of a robot furthest from itself. This function does not solve POINT, since of there are only two robots and their clocks are synchronized, then they simply continue to exchange their positions. In fact, using an argument similar to that in the proof of Theorem 1, we can prove the following theorem. Let $\psi_{point1}(P_r, p^*) = \{bp^* + (1-b)p\}$, where b is a constant real number.

Theorem 2 [2] *Function ψ_{point1} correctly solves problem POINT iff $0 < b < 1$.*

Another possible algorithm for POINT is function $\psi_{point2}(P_r, p^*) = \{g\}$, where g is the gravity center of p^* and the points in P_r. (We can prove its correctness using an argument similar to that in the proof of Theorem 1.) Note that if the local clocks are synchronized, then it can solve POINT in one step, since all robots simultaneously move to g. However, if the clocks are not synchronized, it is not clear whether or not ψ_{point2} can solve POINT more quickly than ψ_{point} in general.

Function $\psi_{point3}(P_r, p^*) = \{(p^* + q)/2\}$, where q ($\neq p^*$) is a point in P_r closest to p^*, moves robot r to the midpoint of r and a robot nearest to r. This function may seem to be a better algorithm than ψ_{point}, since finding a nearest robot can be easier than finding a furthest one. Unfortunately, ψ_{point3} does not solve POINT. To see this, consider the case of four robots

---

[2] The theorem is partially suggested by Saito[4].

r_1, r_2, r_3 and r_4 in which initially, r_1 and r_2 (or r_3 and r_4) are close to each other but r_1 and r_3 are far apart. Then, r_1 and r_2 (or r_3 and r_4) converge to a point, but since this process never terminates in finite steps, the four robots never converge to a point.

The next theorem states that any problem solvable by a deterministic algorithm is actually solvable by an algorithm for POINT, such as ψ_{point}.

Theorem 3 *Let π be a predicate which is invariant under any motion and similarity. Then there is a deterministic algorithm for solving the convergence problem for π if and only if for any P, $\pi_{point}(P)$ implies $\pi(P)$.*

The proof of Theorem 3 uses the fact that "clones" remain indistinguishable forever, where two robots r and s are *clones* of each other if they have the same coordinate system (and thus by assumption, the same initial position), and their local clocks are synchronized. In the following, we assume that no clones exist.

4 Agreement on a Coordinate System

In this section we investigate the problem of obtaining a common coordinate system, where the goal is to let the robots agree on the origin, unit distance, and direction. If the robots have a common coordinate system, then a special leader robot can be elected (if necessary) that controls all other robots. It turns out, however, that agreeing on a coordinate system is not possible in general. Specifically, we show that, while it is possible for the robots to agree on the origin and unit distance, agreeing on direction is not possible in general. We also discuss a limitation of oblivious algorithms.

4.1 Agreement on the Origin

We call the problem of agreeing on the origin ORIGIN. We first consider the formation problem for *two* robots, called MEET, for predicate π_{point} introduced in Section 3. The goal of MEET is thus to move two robots to a single point *in finite steps.* Recall that function ψ_{point} of Section 3 solves the corresponding convergence problem for any number of robots. We have:

Theorem 4 *There is no oblivious algorithm to solve problem MEET.*

Outline of Proof Consider robots r and s. Suppose that there is an oblivious algorithm ψ that solves MEET. We first observe that, regardless of Z_r and Z_s, there must be positions p and q (given in terms of Z) of robots r and s, respectively, such that, from that configuration, ψ moves exactly one of r and s to the position of the other. (If this is not true, then by changing the rate of their local clocks carefully and using the fact that ψ is oblivious, we can obtain an infinite sequence of moves that never brings the robots to a single point.) Now, consider an initial configuration in which, in terms of Z, r is at $(0,0)$, s is at $(1,1)$, $Z_r = Z$, and Z_s is obtained from Z_r by translating its origin to $(1,1)$ and then rotating it about $(1,1)$ for 180 degrees. Then the situation looks identical to both r and s. Thus even though ψ may not be deterministic, all future configurations can look identical to r and s, if their clocks are synchronized and they always move in the same (symmetric) manner. So it is possible that they never reach a configuration in which ψ moves exactly one of r and s to the position of the other. This is a contradiction. □

On the other hand, we can show that there is a non-oblivious algorithm, called ψ_{meet}, for solving MEET. Instead of describing ψ_{meet} formally, we explain the behavior of a robot that executes it.

Algorithm ψ_{meet}: When robot r becomes active for the first time, it tests whether the other robot s is at the same position as itself. If so, then r does not do anything. Otherwise, r rotates its coordinate system Z_r about the origin so that s is on the positive y-axis of Z_r, say at $(0, a)$. (By assumption, r is at $(0, 0)$ of Z_r.) Then it moves in the positive x direction of Z_r, over any nonzero distance. It then continues to move in the same direction whenever it becomes active, until it observes that the position of s has changed twice.

Now, r knows line ℓ that contains the trajectory of s. (Note that by symmetry, ℓ is the x-axis of s's coordinate system Z_s.) If ℓ is parallel to the x-axis of Z_r, then r moves to $(0, a/2)$. Otherwise, r moves to the intersection of ℓ and the x-axis of Z_r. □

Theorem 5 *Function ψ_{meet} correctly solves problem MEET.*

Outline of Proof The key observation is the following: When one of the robots, say r, observes that the position of s has changed twice, s must have already observed that r's position has changed at least once, and thus s knows where the x-axis of Z_r is. Similarly, s will know that r knows where the x-axis of Z_s is. Now, it is easy to show that the trajectory of r (i.e., the x-axis of Z_r) and the trajectory of s (i.e., the x-axis of Z_s) are parallel iff r and s become active for the first time simultaneously. Also, if they become active for the first time simultaneously, then each robot has seen the other robot when it was at its initial position. So if the two trajectories are parallel, then they move to the midpoint of their initial positions. (They know where that point is.) Otherwise they move to the intersection of the two axes. □

Note that in ψ_{meet}, both robots know the position where they meet before reaching there. (This is not true for the algorithms in the previous section.) Thus ψ_{meet} solves ORIGIN for two robots, except in the case when the robots have the same initial position. In this case, neither of the robots ever moves, and thus they will never know whether they have reached an agreement. We cope with this problem by using additional instructions before ψ_{meet}: Each time r becomes active and finds that the other robot s is at the same position as itself, it moves over distance 1 in the positive x-direction of Z_r. (Robot s moves in a similar manner, using its local clock c_s and coordinate system Z_s.) Then, using the assumption that clones do not exist, we can show that eventually, r and s will occupy different positions. Then each of them starts executing ψ_{meet} when it becomes active again, and eventually they both move to the same position (and agree on the origin). So ORIGIN is solvable for $n = 2$.

Now, we describe an algorithm ψ_{origin} for ORIGIN for the case $n > 2$. Before we apply ψ_{origin}, we use another algorithm $\psi_{scatter}$ to transform the given initial distribution of the robots into one in which (1) no two robots occupy the same position, and (2) the robots are not located on a single line segment. Algorithm $\psi_{scatter}$ consists of two parts. Part 1 moves all the robots to distinct positions, and is similar to, but more complex than, the scattering process we used for ψ_{meet}. (We can prove its correctness using the assumption that no clones exist.) Part 2 assures that the robots do not form a single line segment.

Algorithm $\psi_{scatter}$:
Part 1: Suppose that robot r becomes active and finds that not all robots occupy distinct positions. (Otherwise, Part 1 is over.) Let P_r be the positions of the robots that r observes, given in terms of Z_r. (1) If no other robot is located at the current position of r, then r does not move, until it observes that all robots occupy distinct positions. (2) On the other hand, if $m \geq 1$ other robots are located at the current position of r, then r finds the distance $a_r > 0$ to its nearest neighbor (excluding those at the current position of r), and computes the value $b_r = a_r f(a_r)/2$, where for $x \geq 0$, $f(x) = 1 - 1/2^x$ is a monotonically increasing function in the range $[0, 1)$. Then r moves over distance $b_r/2$ in the positive x-direction. After that, each time r becomes active and finds that there are still m other robots at the same position as itself, r moves over distance $b_r/2^k$ in the same direction, if this is the k-th move ($k = 2, 3, \ldots$). When r eventually becomes active and finds that there are fewer than m other robots at the same position as itself, r repeats this entire procedure from the beginning of (2), using a new value of a_r (and resetting k). This process is repeated each time r finds that fewer robots are located

at the same position as itself, until no other robot is found to occupy the same position as r. Then r waits, without moving, until all robots occupy distinct positions.

Part 2: (At this moment, all the robots occupy distinct positions.) Suppose that robot r becomes active and finds that the robots are located on a single line segment. (Otherwise Part 2 is over.) If r is located at an endpoint of the segment, then it does not move. If r is not at an endpoint, then r moves over any distance in the direction perpendicular to the segment. (As soon as some robot does this, the robots no longer form a single line segment, and of course, all the robots occupy distinct positions.) □

One technical difficulty is that in general, a robot cannot determine, given the positions of the robots observed at two time instants t_1 and t_2, which robot has moved to which position between t_1 and t_2. We overcome this problem by imposing a bound on the maximum distance that any robot can move while other robots are inactive, so that the robot at position p at time t_1 must be at the position closest to p at time t_2. Specifically, when robot r becomes active for the first time, it memorizes the distance $a_r > 0$ to its nearest neighbor. Then r moves at most distance $a_r\epsilon/2^k$ in the k-th move, where $0 < \epsilon \leq 1/2$ is a constant chosen by the algorithm. This restriction assures that r remains in the interior of the $a_r\epsilon$-neighborhood of its initial position. Since the interiors of such neighborhoods of two robots located at different positions do not intersect, any robot can correctly know the position of r, even after it remains inactive for a long time. In Part 2 of $\psi_{scatter}$, we used a similar technique to prevent two robots located at different positions from moving to the same position.

Algorithm ψ_{origin} is given next. Each robot first executes algorithm $\psi_{scatter}$, and then starts executing ψ_{origin} as soon as it finds that (1) no two robots occupy the same position, and (2) the robots are not located on a single line segment. So in the following description of ψ_{origin}, we assume that these two conditions are already satisfied.

Algorithm ψ_{origin}: Let P_r be the positions of the robots that r observes when it becomes active for the first time (after finishing $\psi_{scatter}$). Robot r finds the distance a_r to its nearest neighbor. If r is not at a corner of the convex hull $CH(P_r)$ of the points in P_r, then r memorizes the current position p of its nearest neighbor, and moves towards p each time it becomes active, staying in the interior of the $(a_r/2)$-neighborhood of its initial position. (See the explanation in the preceding paragraph.) Suppose that r is at a corner of $CH(P_r)$, and let a, b, c and d be consecutive corners in clockwise order, where r is at b. Then r memorizes the direction that is away from a along the line containing $\overline{ab}$, and moves in that direction each time it becomes active, staying in the interior of the $a_r\epsilon$-neighborhood of b for some $\epsilon \leq 1/2$. Here, ϵ is chosen so that the $a_r\epsilon$-neighborhood does not intersect the line containing $\overline{cd}$. (This assures that the robot s at corner c remains to be at a corner of the convex hulls of the new positions of the robots even after r and s move.)

Robot r continues to move as described above, until it observes that the position of each robot has changed at least twice. Then, r knows line ℓ_s containing the trajectory of s for each robot s at a corner of the initial convex hull. Since the convex hull of the initial positions of the robots is a k-sided polygon for some $k \geq 3$, the lines $\{\ell_s\}$ determine a unique, smallest convex polygon Q that contains the initial convex hull. So r chooses the center of gravity of the corners of Q as the (common) origin. □

We can show that using ψ_{origin}, every robot eventually knows the lines $\{\ell_s\}$ correctly. (The trajectories of the robots that are not at a corner of the convex hull are not used to obtain the common origin. We must still move such robots, however, in order for other robots r to know that they have become active sufficiently many times and observed r's movement.) So Algorithm ψ_{origin} solves ORIGIN for $n > 2$.

The following theorem summarizes the discussion given above.

Theorem 6 *ORIGIN is solvable for any $n \geq 2$.*

4.2 Agreement on the Unit Distance

We call the problem of agreeing on the unit distance UNITDIST. For the case $n = 2$, we can show that in ψ_{meet}, each of robots r and s eventually finds the initial position of the other. Then they can use the distance between their initial positions as the unit distance. For the case $n > 2$, the robots can choose, as the unit distance, the length of a shortest side of the k-sided convex polygon Q that they obtained using algorithm ψ_{origin} in Subsection 4.1. So we have:

Theorem 7 *UNITDIST is solvable for any $n \geq 2$.*

4.3 Agreement on Direction

Now we show that the third problem of agreeing on direction is unsolvable in general. Let us call this problem DIRECTION.

Theorem 8 *There is no algorithm for solving DIRECTION, even if $n = 2$ and the local clocks are synchronized.*

Outline of Proof Consider two robots r and s such that (1) Z_s is obtained from Z_r by rotating it for 180 degrees about the origin, and (2) the local clocks of r and s are synchronized. Then it is possible that r and s move in the same (symmetric) manner all the time, and thus when r decides the direction of its positive x-axis, s chooses the opposite direction for its positive x-axis. □

Note that as we stated in Section 2, the other two agreement problems are trivially solvable if the local clocks are synchronized.

5 Robots with a Sense of Direction

We say that the robots have a sense of direction if they agree on the direction of the positive x-axis. The main result of this section is the following.

Theorem 9 *If the robots have a sense of direction, then they can agree on a common coordinate system, and elect a leader.*

Although the first claim follows immediately from the discussions in Subsections 4.1 and 4.2, in the full paper we show that the following algorithm, called $\psi_{coordinate}$, solves *both* the coordinate agreement problem and the leader election problem, if the robots have a sense of direction. As before, it is assumed that we first use $\psi_{scatter}$ so that no two robots will occupy the same location.

Algorithm $\psi_{coordinate}$: We only give an outline. (We use "east" to denote the positive x-direction, etc.) The idea is to choose the horizontal line though the northern-most robots as the common x-axis, and the vertical line though the eastern-most robots as the common y-axis. To achieve this goal, (1) each eastern-most robot that is not northern-most continue to move north within a "small" neighborhood of its initial position (as explained in Section 4), and (2) all other robots continue to move west. (If there is a robot that is both eastern-most and northern-most, then it first moves to a point to the south of any robot that is eastern-most, and then does (1).) Then eventually, every robot knows the trajectories of all other robots, and chooses the northern-most (or eastern-most) trajectory as the common x- (or y-) axis, The unit distance is then chosen to be the minimum distance > 0 between any two westbound (i.e., horizontal) trajectories. The leader is the northern-most robot among the eastern-most robots. □

Since the leader can compute the final positions of all the robots that satisfy any given predicate and "guide" them to their respective final positions, we obtain the following theorem. Details will be found in the full paper.

Theorem 10 *If the robots have a sense of direction, then for any predicate π, the convergence and formation problems for π is solvable.*

In particular, we have:

Corollary 1 *If the robots have a sense of direction, then the problems of forming a circle and a line segment are solvable.*

6 Concluding Remarks

We viewed a group of mobile robots as a distributed system in which the components can communicate with each other only by means of their moves, and investigated the possibility and impossibility of solving some of the problems related to the formation of geometric figures in the plane. Our study indicates that the assumptions we make on the knowledge and capabilities of the robots can affect the difficulty of solving the given problem in a subtle way. We are currently conducting similar investigations on (1) randomized algorithms, (2) the case in which the motion of a robot is not instantaneous, and (3) the 3-dimensional case. The results will be reported in a future paper.

References

[1] K. Fujimura, "Model of reactive planning for multiple mobile agents," *Proc. IEEE Int. Conf. on Robotics and Automation*, Sacramento, CA, June 1991, pp. 1503–1509.

[2] T. Fukuda and S. Nakagawa, "Approach to the dynamically reconfigurable robot systems," *Journal of Intelligent and Robotics Systems 1*, 1988, pp. 55-72.

[3] N. Hirota, *private communication,* 1992.

[4] Y. Saito, *private communication,* 1992.

[5] K. Sugihara and I. Suzuki, "Distributed motion coordination of multiple mobile robots," *Proceedings of the 5th IEEE International Symposium on Intelligent Control*, Philadelphia, Pennsylvania, September 1990, pp. 138–143.

[6] O. Tanaka, *private communication,* 1992.

[7] J. Wang and G. Beni, "Cellular robotic systems: Self-organizing robots and kinetic pattern generation," *Proc. of the 1988 IEEE International Workshop on Intelligent Robots and Systems*, Tokyo, Japan, 1988, pp. 139–144.

[8] Yamashita, M., and Kameda, T., "Computing on anonymous networks," *Proc. 7th ACM Symposium on Principles of Distributed Computing,* 1988, 117–130.

The Changing Mission of Engineering Education

Chalmers F. Sechrist*
National Science Foundation
Division of Undergraduate Education
4201 Wilson Boulevard
Arlington, VA 22230

Introduction

The role of the engineer in society is changing in response to the diverse needs and demands of a dynamic and global workplace. Increased knowledge of design, manufacturing, marketing, economics, the business world, international affairs, and the humanities will become increasingly important. Education of the whole person and preparation of engineering students for leadership in non-traditional careers will be high-priority objectives in the future.

Therefore, there is a need to revitalize the content, pedagogy, and quality of undergraduate education in mathematics, the sciences, technology, and engineering; to enhance interest in these disciplines among students entering colleges and universities; to increase the participation of underrepresented groups; and to encourage capable students to prepare themselves for teaching at the college and pre-college level. These challenges require new and innovative approaches to all aspects of the undergraduate learning experience, but especially to introductory-level curricula, courses, and laboratories.

Examples of some NSF-supported programs are mentioned including the areas of faculty enhancement, instrumentation and laboratory improvement, and course and curriculum development.

---

* On leave from the University of Illinois at Urbana-Champaign, College of Engineering, Department of Electical and Computer Engineering

Future Engineering Curricula

Engineering curricula of the future will be designed in response to the skills, knowledge, abilities, and attitudes that engineers of the 21st century must possess in order to function effectively in a world whose socio-political-economic nature is changing.

The problems that will dominate the engineering profession in the future will require people skills to define the problems, and sensitive multicultural leadership talent to lead a team of experts, not all of them engineers, to solve the problems within an open, global community.

Engineers in the 21st century may be confronted with making total system decisions that affect not only many branches of engineering but also social and political issues---issues that are especially important to civil infrastructure considerations and global competitiveness.

Of major importance is the realization that engineering education must be reinvented in a form that can help the U.S. manufacturing and service industries re-establish a competitive position in worldwide markets. Engineering education must change significantly to support the new quality-oriented environment in U.S. industry which is adopting the principles of customer focus, continuous improvement, and concurrent engineering, in order to survive and flourish in the global economy. An important component in future engineering curricula should be managerial-type decision making in the context of real world engineering practice.

To meet the demands of this new dominant mode of engineering practice there is required not only a change in the undergraduate curriculum, but a change in the culture of the engineering education enterprise. A new paradigm shift is needed in engineering education that encompasses the solid mathematical and scientific base of current curricula, and introduces engineering subjects in ways that actively engage students in team efforts to address design problems typical of a quality-oriented industrial environment. This implies a much stronger project and case-study orientation, with interdisciplinary, cross-functional teams of engineering and business students working on projects that integrate design, development, manufacturing, and marketing of quality products and processes. Engineering sciences would be introduced as needed to support the project, but as a means to an end, not in isolation as ends in themselves.

The 21st Century Engineer

The attributes or characteristics of future engineers will be determined on the basis of the skills, capabilities, and attitudes needed in the global workplace. Thus, future curricula should prepare an engineering graduate who is:

1. able to specify, design, develop, and prepare for production, cost- efficient, state-of-the-art products which will compete for value with anything in the world
2. competent in projects that integrate design, development, manufacturing, and marketing of quality products and processes
3. knowledgeable about manufacturing, the environment, marketing, economics, the business world, international affairs, the humanities, political and social issues, continuous improvement, concurrent engineering, and managerial decision making
4. a creative problem solver and designer (including societal problems)
5. a cross-functional team player with interdisciplinary competence
6. comfortable with uncertainty, ambiguity, open-ended thought, and change
7. flexible, adaptable, and career ready
8. prepared to enter non-traditional careers
9. the "bridge" spanning the culture of the scientist and the culture of the humanist
10. a good communicater, with interpersonal and management skills; and with sensitive and multicultural leadership talent

NSF-Sponsored Curriculum Innovation in Undergraduate Engineering Education

The National Science Foundation (NSF) provides the leadership and support for the Nation's efforts to improve science, mathematics, engineering, and technology education. The Division of Undergraduate Education (DUE) serves as the focal point for NSF's agency-wide effort in undergraduate education. DUE's programs and leadership efforts aim to strengthen and ensure the vitality of undergraduate education in science, mathematics, engineering, and technology for all students as:

* science, mathematics, or engineering majors
* students in science and engineering technology programs
* future teachers at the elementary and secondary school level
* non-science majors seeking scientific and technical literacy

DUE programs enhance the quality of instruction in the diverse institutions of higher education: two-year colleges, four-year colleges, and universities. Emphasis is placed on improving access for all segments of U.S. society, including populations underrepresented in science, mathematics, and engineering studies and in technical and teaching careers. The programs briefly described below encompass most of the engineering activities supported by DUE.

Calculus and the Bridge to Calculus. This activity supports projects to improve the first two years of the calculus curriculum, including linear algebra and differential equations. The program has been broadened also to include the preparation for calculus.

Instrumentation and Laboratory Improvement. This program supports projects designed to generate new and improved approaches to laboratory and field-based instruction. It consists of two components: The **Instrumentation** component provides matching funds for the purchase of equipment needed to make possible improvements in undergraduate laboratories at any level; and the **Leadership in Laboratory Development** component provides resources, including time, technical support, and travel necessary for projects that have the promise of being national models for laboratory instruction.

Undergraduate Faculty Enhancement. This program supports activities that help faculty members primarily engaged in the instruction of undergraduates to gain experience with recent advances in their fields, with new experimental techniques, and with ways of incorporating these into undergraduate instruction. Projects are regional or national in scope and typically consist of hands-on short courses or workshops, along with follow-up activities that encourage sustained interaction among participants. A major component is regional coalitions between two-year and four-year institutions.

Leadership Opportunity in Science and Humanities. The NSF, the National Endowment for the Humanities' Division of Education, and the Department of Education's Fund for the Improvement of Post Secondary Education (FIPSE) jointly fund this program which supports activities for the development of undergraduate courses and curricula that link meaningfully the study of science and the humanities.

Undergraduate Course and Curriculum Development in the Sciences, Mathematics, and Engineering. The goals of this program are: to encourage development of courses and curricula to meet the Nation's need for high quality scientists, engineers, mathematicians, and technicians, dedicated and able teachers at the pre-college and college levels, and scientifically and technically literate citizens; and to inspire faculty to devote creative energy to educational activities. The program currently emphasizes the introductory level courses for majors and non-majors. Grants provide for the planning, implementation, assessment, and dissemination of projects of potentially national scope, which are designed to improve curricula and the learning environment and to develop new courses, materials, software, and technologies. The forerunner of this program was the Undergraduate Curriculum Development Program in Engineering, which will be reviewed in the next section of this article.

Engineering Education Coalitions (EECs). The EECs program was developed to stimulate bold, innovative, and comprehensive models for systemic reform of undergraduate engineering education. The EEC program is funded by the Engineering Education and Centers Division in the Engineering Directorate of NSF. The purpose of the program is to join universities and colleges of diverse characters in collaboration to experiment and implement, acting as change agents for the engineering education community at large. The first four EECs are: ECSEL, with design across the curriculum as a theme; SYNTHESIS, with themes of Synthesis of knowledge for problem-solving, and National Engineering Education Delivery System; GATEWAY, with the theme of opening new "gateways" for learning by altering engineering education to an integrative process; and SUCCEED which aims to develop "Curriculum 21" emphasizing the process of engineering and the engineering education process. Two new EECs will be funded in 1993: FOUNDATION which aims to design

"Foundation" curricula; and the coalition for new manufacturing education, which will develop curriculum modules for multimedia delivery in a manufacturing environment.

The views and opinions expressed in this extended abstract are those of the author and do not necessarily reflect the positions or opinions of the National Science Foundation.

MECHATRONIC SYSTEM DESIGN AT RENSSELAER

Dr. Kevin Craig
Department of Mechanical Engineering, Aeronautical Engineering & Mechanics
Rensselaer Polytechnic Institute
Troy, NY 12180
Phone: 518-276-6671 E-Mail: Kevin_Craig@meche.rpi.edu FAX: 518-276-2623

ABSTRACT

What constitutes a winning design or a successful designer? The two-course, senior-elective sequence in *Mechatronics* at Rensselaer addresses these two issues and is described in this paper. Mechatronics, as an engineering discipline, is the synergistic combination of precision mechanical engineering, electronics, control engineering, and computer science, all integrated through the design process. Starting at design and continuing through manufacture, mechatronic designs optimize the available mix of technologies to produce quality precision consumer products. If winning designs are to be produced in today's world, it is imperative that electronics and computer control be included in the design process at the same time the basic functions and properties are defined. The real benefits to industry of a mechatronic approach to design are shorter development cycles, lower costs, and increased quality, reliability, and performance. The first course, *Mechatronics*, covers control sensors and actuators, interfacing sensors and actuators to a microcomputer, discrete and continuous controller design, and real-time programming for control using the *C* programming language. These are the fundamental areas of technology on which successful mechatronic designs are based. Throughout the course mechatronic design principles are stressed and are reinforced through the reverse engineering of successful mechatronic products and systems. This course is heavily laboratory based, with the course material presented so that information discussed in lecture is immediately reinforced in the laboratory. A premium is placed on interactive learning through: student-team formation in the lectures and the discussion of design-related issues, hands-on laboratory exercises involving industrial-quality hardware, computer-aided design involving the latest computer control and design software, and the encouragement of critical thinking throughout the course.

What makes a successful designer? It is clear to engineering educators that practicing design is an integral part of teaching design. After an engineer is taught analytical techniques and component technologies and after an engineer studies the elements of the design procedure, an engineer learns good design methodology through experience. The purpose of the second-semester course, *Mechatronic System Design*, is to let the engineering student experience design. The projects for this course come from industrial sponsors who provide the real-world design problems for the student design teams to investigate as well as any project-specific hardware and equipment needed for successful completion of the design project. The problems and the design experience are real. Through a step-by-step case study of an actual microprocessor-based mechatronic product complete with detailed design drawings and a working prototype, mechatronic design principles are reviewed as are microprocessor hardware, software, and interfacing basics. Each design team generates an outstanding mechatronic conceptual design and then proceeds through detailed design, construction, and testing. The end product of this course is a microprocessor-based working prototype accompanied by a written design report with drawings, schematics, and documentation, and an oral presentation. In short, it is a completed mechatronic product design.

While most of this paper describes instruction in mechatronics at the undergraduate level at Rensselaer, active research in mechatronic system design at the graduate level is critical to ensure that what is being taught reflects the best industrial practices. Research projects in collaboration with the Xerox Mechanical Engineering Sciences Laboratory are currently underway and these are discussed in this paper.

1. The Current Situation

Rensselaer Polytechnic Institute is a private, coeducational university serving approximately 4400 undergraduate and 1600 graduate students with more than 110 programs in the Schools of Architecture, Engineering, Humanities and Social Science, Management, and Science. The underlying goal of these programs is to develop broadly-educated women and men who will be able to exert leadership in society and contribute to human welfare. With 2600 students, Rensselaer's Engineering School offers one of the largest undergraduate engineering programs of any private university in the United States. In addition, Rensselaer in recent years has been consistently ranked among the top 15 engineering research universities in the country.

Undergraduate engineering education at Rensselaer consists of two phases: an interdisciplinary core curriculum, where instructors come from various departments, followed by the disciplinary curriculum implemented by an individual department. The Department of Mechanical Engineering, Aeronautical Engineering & Mechanics at Rensselaer offers bachelors, masters, and doctoral degrees in the three disciplines represented in the name. The department awards over 200 Bachelor, 80 Master, and 17 Doctoral degrees per year, the major portion of which are mechanical engineering degrees.

The current economic situation in which the United States and the world finds itself is a challenge to university education. It is generally agreed that *domestic renewal* must now be the nation's highest priority and that *economic security* must now take precedence over military security for the future well-being of the country. The challenge to engineering educators is to develop courses that are *relevant* to the needs of the nation and its industry. We must give our students the tools to help quickly reverse the economic erosion our country is experiencing.

If we look at successful industrial businesses, we find that these firms not only promote new technology, but are able to bring quality products, that the customer wants, to market in a timely manner using existing technology. Technological superiority is not sufficient for economic prosperity. The nation's leadership in creating new technologies will pay off only when this technology is rapidly implemented and embedded in high-quality, cost-competitive, innovative products that meet or exceed customer expectations.

2. What is Mechatronics?

Mechatronics, as an engineering discipline, is the synergistic combination of mechanical engineering, electronics, control engineering, and computer science. The key element in mechatronics is the integration of these areas through the design process. Starting at design and continuing through manufacture, mechatronic designs optimize the available mix of technologies to produce quality precision consumer products. The benefits of a mechatronic approach to design are shorter development cycles, lower costs, and increased quality, reliability, and performance. All are benefits important to the customer and necessary for the future growth of American industry.

The need for specialization in industry is self-evident; without specialization it is not possible to develop basic technologies. However, as the importance of strengthening the nation's industrial base is realized, the number of tasks requiring knowledge from several fields is steadily growing. Experts in the mechatronics field acquire general knowledge of various interdisciplinary techniques which enable them to master the entire design process. They are also able to use the special knowledge resources of other people and the particular blend of technologies which will provide the most economic, innovative, elegant, and appropriate solution to the problem at hand. Industry needs mechatronics engineers to quickly develop innovative products with performance, quality, and low cost. Industry, government, and universities can and should work together to develop the curriculum necessary to produce successful mechatronics engineers.

3. Mechatronics: Course and Laboratory

The senior elective course *Mechatronics*, was first taught at Rensselaer during the fall 1991 semester to 42 students, and then during the fall 1992 semester to 68 students, and now in the fall 1993 semester to 88 students. Of the 88 students this fall, all are mechanical engineering students and 80% are senior undergraduates, while the rest are graduate students.

The first course, *Mechatronics*, covers control sensors and actuators, interfacing sensors and actuators to a microcomputer, discrete and continuous controller design, and real-time programming for control using the *C* programming language. These are the key areas of technology on which successful mechatronic designs are based. The unifying theme for the course is the integration of these key areas into a successful design. Throughout the course mechatronic design principles are stressed and are reinforced through the reverse engineering of successful mechatronic products and systems. This course is heavily laboratory based, with a 2-hour weekly laboratory in addition to 3 hours of classroom instruction each week. The course material is presented so that information discussed in lecture is immediately reinforced in the laboratory. A premium is placed on interactive learning through: student-team formation in the lectures and the discussion of design-related issues, hands-on laboratory exercises involving industrial-quality hardware, computer-aided design involving the latest computer control and design software, and the encouragement of critical thinking throughout the course. A wide variety of analog and digital sensors and actuators are used in the laboratory exercises. A list of the course lecture topics, laboratory exercises, and references is given in the appendix.

4. Mechatronic System Design: New Course and Laboratory

From the above description of the existing course, *Mechatronics*, it is obvious that the fundamental elements of mechatronics are covered, but time and laboratory facilities make it very difficult to go through the actual "design and build" aspects of a mechatronic system. The real benefits to industry of the integrating approach of mechatronics is achieved in design. A second semester course and laboratory that can focus completely on mechatronic design addresses this aspect. The major emphasis of this course is microprocessor-based detailed design from concept through construction and testing.

Is there a need for a two-course sequence in mechatronics? The answer to this question relates to what constitutes a winning design or a successful designer. The concept of mechatronics as an integrating approach to engineering design arose from the need for the timely development of products and processes with a greater range of performance options than present-day conventional systems. Mechatronics was an inevitable natural progression as computers became more common and less expensive. Virtually every new mechanical design (e.g., in consumer products, laboratory instrumentation, factory automation, aerospace systems) can now consider a microprocessor subsystem as part of the possible solution. Old designs can now be reevaluated to determine where electronics and computer control can be incorporated, if the product is to remain competitive in today's market. Indeed, it is imperative that electronics and computer control are included in the design process at the same time the basic functions and properties are defined. And so the answer to the opening question is YES, the need is great if winning designs are to be produced. The second issue involved in this question is what makes a successful designer. It is clear to engineering educators that practicing design is an integral part of teaching design. After an engineer is taught analytical techniques and component technologies and after he/she studies the elements of the design procedure, an engineer learns good design methodology through experience. From this perspective, the answer to the opening question is also YES. The purpose of this second semester course is to let the engineering student experience design.

Student design teams should have a real customer, i.e., industrial supporters of this laboratory. The projects for this course come from industrial sponsors (e.g., Xerox and Ford) and health-related organizations (e.g., Albany Center for the Disabled) who provide real-world design problems for the students to investigate as well as any project-specific hardware and equipment needed for successful completion of the design project. The problems and thus the design experience are real.

Mechatronic System Design will be a limited enrollment (30 students maximum) senior elective course offered every spring semester (beginning in the 1994 spring semester) with the primary prerequisite *Mechatronics*, which is offered every fall semester. Evaluation criteria will be developed for selection of these students with the intention that the best graduates from *Mechatronics* will be selected. The two-course sequence in mechatronics fits very well in the current mechanical engineering undergraduate curriculum at Rensselaer. *Mechatronics* can now be used either as a technical elective or an experimental methods elective, and *Mechatronic System Design* could be used either as a technical elective or as a substitute for the mechanical engineering capstone design course.

The *Mechatronic System Design* course will consist of a one-hour weekly lecture and two, two-hour weekly laboratory periods. Students will be divided into three-person design teams for the laboratory and design project. The first third of the course will focus on mechatronic design principles and microprocessor hardware, software, and interfacing. During this time, the lectures will cover the systematic approach to mechatronic design, i.e., problem statement, state-of-the-art review, specifications list and identification of key specifications, generation and evaluation of design concepts, selection of the outstanding mechatronic conceptual design, detailed design, construction and testing, documentation and presentation. This approach will be reinforced with a step-by-step case study of an actual microprocessor-based mechatronic product complete with detailed-design drawings and a working prototype. At the same time that these steps are discussed and illustrated, the three-person student design teams will be performing these steps through the conceptual design phase for their industry-sponsored design problem. An in-class examination covering the fundamental principles of mechatronic design as demonstrated and discussed will be given to the students. In the laboratory periods during this first third of the course, the three-person design teams will learn microprocessor hardware, software, and interfacing by redoing one of the *Mechatronics* laboratory experiments with a microprocessor rather than a microcomputer.

The remaining two thirds of the course will be devoted to the detailed design, construction, and testing of the team's conceptual design. During this time, both lecture and lab periods will be used for this work and lecture topics will be introduced as needed depending on the real world design problem selected. Available company literature will also be discussed (e.g., market share, product base, market performance) so that student teams can appreciate how their design will fit the company's needs. Weekly meetings between the instructor and each of the teams will be held at which weekly progress reports will be submitted. At approximately two-thirds of the course, the student teams will convene for a preliminary design review. The teams will present material on available hardware, preliminary test data, design approaches, etc. Experiences will be shared with other teams as the full breadth of design solutions can be explored. Teams can trade "war stories" so that the design experience of the total class can be enriched.

The ultimate goal of the course is to have a working prototype accompanied by a written design report including drawings, schematics, and documentation. An oral presentation by each of the design teams with the industrial sponsor present will also be a requirement. The total grade for the course will consist of: design examination 15%, microprocessor laboratory experiment 15%, preliminary design review 10%, working prototype 30%, written report 20%, and oral presentation 10%.

5. Innovative Aspects of the Two-Course Sequence in Mechatronics

There are many innovative aspects to the existing *Mechatronics* course as well as to the new *Mechatronic System Design* course:

Mechatronics:

1) Integrating the study of control sensors and actuators, interfacing sensors and actuators to a microcomputer, discrete and continuous controller design, and real-time programming for control using *C*, all in a single mechanical engineering undergraduate course.
2) Development of a well-integrated mechatronics laboratory with industrial-quality hardware to give the students hands-on experience using a microcomputer to control actual physical systems.
3) Use of the most up-to-date computer software for dynamic system modeling and analysis, control system design and simulation, and real-time programming.
4) Use of mechatronic design principles to unify the course, e.g., reverse engineering of several mechatronic products and systems throughout the course.
5) Demonstration and discussion of the latest technology, e.g., active magnetic bearings, active materials and smart structures, neural networks, fuzzy logic.

Design of Mechatronic Systems:

1) In-depth discussion of the systematic approach to a mechatronic design and an actual step-by-step demonstration of the approach through a working prototype of a mechatronic product.
2) Study of microprocessor basics, i.e., hardware, software, and interfacing, and immediate implementation in a microprocessor-based mechatronic experiment.
3) Team mechatronic design project with an industrial sponsor/customer with the result being an actual microprocessor-based working prototype. This gives the students real-world design experience.
4) Development of leadership, interpersonal, and both written and oral communication skills throughout the entire team design project.

6. Impact and Value of the Two-Course Sequence in Mechatronics both Inside and Outside Rensselaer

This two-course sequence in mechatronics at Rensselaer addresses the needs of our nation in design and manufacturing and the need for relevance in the undergraduate mechanical engineering curriculum, and will have a tremendous impact both at Rensselaer, the surrounding communities, and nationwide at other engineering universities.

1) At Rensselaer, this two-course sequence in mechatronics gives the undergraduate hands-on design experience with industrial-quality hardware in a real-world design setting. This is most relevant to their future careers in industry.
2) Technology transfer to Hudson Valley Community College (HVCC) will take place through instructor tutorials, transferred laboratory and experimental procedures, shared resources, and student interaction.
3) Similarly technology transfer to local industry by retraining employees in this new technology in cooperation with HVCC will be started. For example, the first third of the new course could be used as a short course for industry.
4) While other universities have similar efforts in mechatronics, e.g., Smart Product Design at MIT, these courses are graduate courses and to our knowledge there is no other undergraduate engineering program in mechatronics as well developed and planned as this one. This makes our effort ideal as a model for implementation at other universities.
5) The motivation for this program development has been the research of the faculty. This program can be the logical incubator for longer term industrial-sponsored research projects.
6) This program in mechatronics supports the Rensselaer strategic initiatives in manufacturing, materials, and design.

7. Mechatronic System Research

While most of this paper describes instruction in mechatronics at the undergraduate level at Rensselaer, active research in mechatronic system design at the graduate level is critical to ensure that what is being taught reflects the best industrial practices. Research projects in collaboration with the Xerox Mechanical Engineering Sciences Laboratory are currently underway. Two examples of this collaboration are:

> *Design and Development of a Flexible, Automated, Fixturing Device for Manufacturing*: Improved workpart fixturing, a major hurdle to overcome in today's competitive, computer-integrated manufacturing environment, has resulted in the need for an adaptable, computer-controlled, high-precision fixture. Based on specifications derived from an actual industrial process, a computer-controlled hydraulic positioning mechanism was designed and built as a proof-of-concept device. The device consists of two hydraulic cylinders, each connected to ground with a revolute joint. The positions of the two cylinders are also connected at the ends with a revolute joint, giving the overall mechanism plane-positioning capabilities (two degrees of freedom). The cylinder lengths are measured directly with linear incremental optical encoders and fed back to a microcomputer. The microcomputer then actuates two, three-position valves to move the cylinders and control the end position. Without using an expensive hydraulic servovalve, the device was able to repeatably control end-point position to less than 0.001".
>
> *Magnetically Suspended Apparatus for Rheological Testing*: It is important to be able to experimentally characterize the parameters of certain viscoelastic materials, ultraviolet-curable polymers, electrorheological fluids, and other fluids of engineering interest under the proper operating conditions for use in industry. There is no commercially-available apparatus capable of performing these measurements with the accuracy and range of shear forces, shear rates, and shear strains desired. The mechatronic concept for such a device, which is currently being built, is described as follows. The apparatus consists of two optical flats, one of which is fixed to the base (ground) with the other mounted in a ferromagnetic plate. The position of the plate relative to ground in three degrees of freedom is sensed with three capacitive or eddy current sensors. The position of the plate above ground is controlled with three electromagnetic actuators. Each actuator will consist of two horseshoe-shaped pole pieces wound with wire and mounted above and below the plate. Energizing the coil on the upper pole piece will cause the ferromagnetic plate to move up (normally away from the fixed plate) while energizing the other coil will cause the plate to move down (normally toward the fixed plate). This electromagnetic suspension will precisely maintain the position of one plate relative to the other. To sense parallelism, two lasers (operating at different wavelengths) and a set of three photodetectors will be used. The laser light will be directed through the two optical flats. If the flats are not parallel, an interference pattern will be produced at the photodetectors. The reference position for the electromagnetic suspension will be adjusted (via computer control) to eliminate this pattern. Once this is done, the suspension will lock onto the analog position from the photodetectors. Using this scheme, achieving parallelism to less than a wavelength of the laser light is expected to be achieved. The second laser will be used in conjunction with the first to determine the gap between the plates accurately. To move the plates normal to each other, the coil currents in the electromagnets of the suspension will be varied. Small controlled normal motions will be possible. To move the plates tangential to each other, a set of linear actuators and position sensors will be used. Any non-tangential component of the force from these actuators will be rejected by the electromagnetic suspension. The stress components will be measured with a tri-axial force transducer connected between the fixed plate and ground. Force transducers will also be inserted between the linear actuator and the plate. A secondary measurement of force is possible by monitoring the currents in the electromagnetic suspension and linear actuators (after proper calibration).

Acknowledgement

The author wishes to acknowledge the following support: a grant from the AT&T Foundation for equip the Mechatronics Laboratory; a grant from the 1992 National Science Foundation Instrumentation and Laboratory Improvement / Leadership in Laboratory Development Program to develop the course *Mechatronic System Design* and further equip the Mechatronics Laboratory; a mechatronics research grant from the Xerox Foundation.

References

1) Craig, K. (1991), "Computer-Aided Control System Design Environment," 1991 IFAC Conference on Advances in Control Education, June 24-25, 1991, Boston, MA, pp. 234-239.

2) Craig, K. (1991), "Controlled Mechanical Systems: Computer-Aided Engineering Environment to Support a New Senior Elective Course," 1991 ASEE Annual Conference, Mechanics division, June 16-20, 1991, New Orleans, LA, pp. 1823-1828.

3) Craig, K. (1992), "Mechatronics at Rensselaer: Integration Through Design," 1992 ASME International Computers in Engineering Conference, August 2-6, 1992, San Francisco, CA.

4) Kurz, K., Craig, K., Wolf, B., and Stolfi, F. (1993), "Design and Development of a Flexible, Automated, Fixturing Device for Manufacturing," 1993 ASME Winter Annual Meeting, Symposium on Mechatronics for Manufacturing, November 28 - December 3, New Orleans, LA.

5) Prakah-Asante, K.O., Islam, A., Walczyk, D. and Craig, K. (1994), "Single-Axis Servomechanism for Control Experiments Involving Joint Friction, Backlash, and Compliance, and Structural Flexibility," Journal of Engineering Design, Vol. 5, No. 1.

APPENDIX:

Lecture Topics, Laboratory Exercises, and References for *Mechatronics*

Lecture Topics:

1. Course Introduction (1 period)
    - ·Motivation for Creating the Course
    - ·Relevance of Course to Current World Manufacturing Situation
    - ·Course Description & Syllabus
    - ·Course Questionnaire
    - ·Reverse Engineering of a Mechatronic Device

2. Introduction to Control Systems (1 period)
    - ·Role of Control Systems in Engineering Design
    - ·Classification of Control System Types
    - ·Basic Benefits of Feedback Control
    - ·Accuracy-Stability Tradeoff in Feedback Systems
    - ·Control System Design Procedures
    - ·Generalized Block Diagram of a Feedback System
    - ·Introduction to Digital Control of Dynamic Systems

3. Measurement Systems (1 period)
    3.1 Types of Applications of Measurement Instrumentation
        ·Monitoring of Processes and Operations
        ·Control of Processes and Operations
        ·Experimental Engineering Analysis
        ·Computer-Aided Machines and Processes
    3.2 Generalized Configurations and Functional Descriptions of Measuring Instruments
        ·Functional Elements of an Instrument
        ·Active and Passive Transducers
        ·Analog and Digital Modes of Operation
        ·Null and Deflection Methods
        ·Input-Output Configuration of Measurement Instruments and Instrument Systems
        ·Methods of Correction for Interfering and Modifying Inputs:
            Method of Inherent Insensitivity
            Method of High-Gain Feedback
            Method of Calculated Output Corrections
            Method of Signal Filtering
            Method of Opposing Inputs

4. Physical & Mathematical Modeling of Dynamic Systems and Dynamic Response (2 periods)
    ·Stages of a Dynamic Investigation
    ·Physical Modeling: From Actual System to Physical Model
    ·Equations of Motion: From Physical Model to Mathematical Model
    ·Simple Physical Models:
        Mechanical, Electrical, Electromechanical, Thermal, Fluid
    ·Analogies between Physical Media
    ·Linearization
    ·Classical Operator Solution Method
    ·Laplace Transform Method
    ·System Frequency Response
    ·Zero-Order Dynamic System Models
    ·1*st* Order Dynamic System Models
    ·2*nd* Order Dynamic System Models
    ·Dead-Time Effects
    ·Loading Effects For Component Interconnection

5. Performance Specifications of Instruments and Component Matching (2 periods)
    ·Transfer Function Models for Sensors/Transducers
    ·Parameters for Performance Specifications
    ·Impedance Characteristics
    ·Instrument Ratings

6. Control Sensors and Actuators (6 periods)
    ·Analog Sensors
    ·Digital Transducers
    ·Stepper Motors
    ·Continuous-Drive Actuators: DC Motors, Hydraulic & Pneumatic Actuators

7. Control System Analysis and Design Review (2 periods)
    Absolute Stability Criteria
        ·Overview
        ·Linear Systems with Constant Coefficients
        ·Routh Stability Criterion
        ·Nyquist Stability Criterion
        ·Root-Locus Interpretation of Stability

- Control System Performance Criteria
    - ·Relations between Practical Economic and Detailed Technical Performance Criteria
    - ·Basic Considerations
    - ·Time-Domain Performance Specifications
    - ·Frequency-Domain Performance Specifications
- Analysis and Design Using Transform Methods
- Analysis and Design Using State-Space Methods

8. Discrete Modeling, System Response, and Discrete Controller Design (6 periods)
    - ·Elements of Discrete Modeling
        - Discrete Process Models
        - Discrete Controller
        - Transfer Functions and Block Diagrams
        - Solution of Difference Equations
        - Stability Analysis
    - ·System Response
        - System Inputs
        - System Response
    - ·Discrete Controller Design
        - Control Algorithms for Desired Closed-Loop Response
        - Sample Period Selection
        - Feedforward Control
        - Cascading Control
    - Analysis and Design Using Transform Methods
    - Analysis and Design Using State-Space Methods

9. Selected Special Topics & Demonstrations (2 periods)
    - ·Magnetic Bearing
    - ·Inverted Pendulum
    - ·Mechanical Positioning Servosystem with Dry Friction, Gear Backlash, Joint Compliance, and Structural Flexibility
    - ·System Identification
    - ·Active Materials as Sensors and Actuators
    - ·Neural Networks
    - ·Fuzzy Logic
    - ·Programmable Logic Controllers

10. Mechatronic Design (2 periods)
    - ·Mechatronic Design: A Systematic Approach
    - ·Mechatronic Design Case Studies

Laboratory Exercises:

1. ·Electronics Review for Mechatronics (3 periods)
2. ·Control Computers, Computer Interfacing, and Familiarization with Data Acquisition Board
3. ·Introduction to *C* Programming and Real-Time Programming for Control
4. ·Introduction to MatLab with Simulink
5. ·Potentiometer and Bar Graph Lab
6. ·Hydraulic Servomechanism with Solenoid-Operated Valves
7. ·Pneumatic Servomechanism with Solenoid-Operated Valves
8. ·Flexible Beam with Strain Gages, Piezoelectric Accelerometer, and Piezoelectric Actuator
9. ·DC Motor Control
10. ·Stepper Motor Control

Textbooks (required):

(1) *Computer Control of Machines and Processes*, J. Bollinger & N. Duffie, Addison Wesley, 1989.
(2) *Control Sensors and Actuators*, C.W. deSilva, Prentice Hall, 1989.

References:

(1) *Real-Time Software for Control*, D. Auslander & C. Tham, Prentice Hall, 1990.
(2) *Design with Microprocessors for Mechanical Engineers*, A.K. Stiffler, McGraw Hill, 1992.
(3) *Control System Principles and Design*, E.O. Doebelin, Wiley, 1985.
(4) *Measurement Systems*, E.O. Doebelin, 4th Edition, McGraw Hill, 1990.
(5) *Art Of Electronics*, Horowitz, P. and Hill, W., 2nd Edition, Cambridge University Press, 1989.
(6) *Linear Control System Analysis and Design*, D'Azzo, J. and Houpis, C., 3rd Edition, McGraw-Hill, 1989.

Hierarchical Control in CIM Education

Lucy Siu-Bik King Ph.D.
Jeffrey Abell Ph.D.
Winston Erevelles Ph.D.

GMI Engineering & Management Institute
I & MSE Department
1700 West Third Ave
Flint, MI 48504

Abstract: In the face of global competition, manufacturing integration is necessary and equipment, process, quality and shop floor operation control play a key role. This paper addresses the shop floor control aspects in a CIM Environment. Hierarchical control is an effective way of shop floor operations management. Efficiency can be increased when used with computer network messaging capability as the main tool for communication from supervisory computer to cell PC to equipment controllers through device drivers. Flowcharting allows for sequential and parallel operations, conditions testing, cross-checking of conditions of all equipment through passing of global parameter values, initializing equipment, downloading, uploading equipment programs and cycle starting a particular piece of equipment. Institutions of higher education must keep pace with these developments by teaching these concepts and practices to engineering students. GMI is responding to this call by setting up a state-of-the-art CIM laboratory for its co-op students.

A. Introduction:

Computer Integrated Manufacturing (CIM) as a driving strategy to reverse declining trends faced by U.S. manufacturers in the global marketplace has gained widespread acceptance in recent years. The Computer and Automated Systems Association (CASA) of the Society of Manufacturing Engineers (SME) has defined CIM as the integration of the total manufacturing enterprise through the use of **integrated systems and data communications** coupled with new managerial philosophies that improve organizational and personnel efficiency [1].

Why CIM Hierarchical Control?

The incorporation of all of an organization's production functions into an integrated enterprise-wide environment gives CIM a global posture in manufacturing competitiveness. These global implications make CIM a very attractive proposition when the accurate and timely exchange of technical and administrative data between organizations and their operating environments has become vital to productivity, profitability, competitiveness, and even survival. To facilitate this data communication, the National Institute of Science and Technology (NIST) defined a structure for data management in the AMRF (Automated Manufacturing Research Facility) CIM control hierarchy [2] (Figure 1).

It is therefore surprising that industrial implementation of CIM has not been very widespread or comprehensive. Obstacles to the implementation of CIM are numerous, and may be caused by philosophical, managerial, financial, and technological shortcomings to mention a few of the areas where deficiencies have been observed. Technological barriers related to data communications and compatibility across multiple platforms have been among the foremost of impediments to the implementation and *control* of CIM, and are of special significance to this paper. A closer examination of these shortcomings that have been cited reveals the lack of adequate and effective education causing a lack of thorough understanding of what CIM control actually means and what its implementation entails. This does not bode well for a philosophy that has been identified as a means to restoring manufacturing competitiveness - both current as well as in the future.

Need for Education

Although CIM technology has evolved considerably from the conceptual descriptions laid out by Harrington, the development of skilled personnel to work in such an environment has been inadequate. The curricula used to teach CIM has proved to be deficient in both scope and content. Experiential education has been repeatedly cited as the key to the corrective measures that may be applied to help the U.S. regain a competitive posture in manufacturing. Both academia and industry have recognized and advocated the need for education and training in order to equip personnel to fit into advanced manufacturing environments - current as well as that of the future [3, 4, 5]. Ideally this facility would provide hands-on training in an environment that would facilitate a seamless transition from the classroom and laboratory to the workplace. Efforts such as Profile 21, Manufacturing 21, and Curricula 2000 have triggered a flurry of industry-based and academic activities seeking to address deficiencies in manufacturing programs across the United States. Educational and research institutions, and industry have sought to develop and implement CIM laboratories and models for the control of environments exhibiting a high level of automation, integration, modularity, and flexibility.

Figure 1:

NIST-AMRF Hierarchy

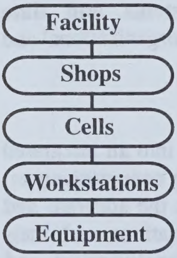
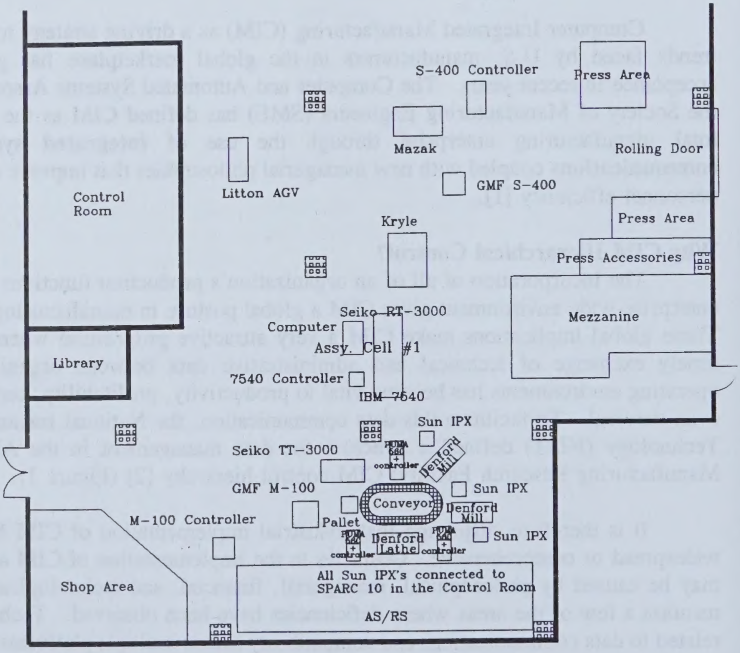

Facility
Shops
Cells
Workstations
Equipment

Figure 2: GMI CIM Lab Layout


S-400 Controller
Press Area
Mazak
Rolling Door
Control Room
Litton AGV
GMF S-400
Kryle
Press Area
Press Accessories
Mezzanine
Seiko RT-3000
Computer
Assy. Cell #1
Library
7540 Controller
IBM 7540
Sun IPX
Seiko TT-3000
PUMA 560 + controller
Denford Mill
GMF M-100
Sun IPX
Conveyor
Denford Mill
M-100 Controller
Pallet
PUMA 560 + controller
Denford Lathe
PUMA 560 + controller
Sun IPX
Shop Area
All Sun IPX's connected to SPARC 10 in the Control Room
AS/RS

CIM education at GMI has been in existence for 4 years now as part of an ABET-accredited program in Manufacturing Systems Engineering. GMI's strong commitment to CIM and effective education in the field at the undergraduate level is underscored by the ongoing development of a separate track within the MSE program devoted to this field. CIM efforts at GMI are supported by an industrial-grade laboratory featuring an AS/RS, an AGV, 5 robots, and 2 CNC machines operating under the DOS environment (Figure 2). Recently, new equipment were purchased under an NSF grant (Grant No. DUE 9251700): 3 new robots, 3 new CNC machines, a safety light fence system, Sun IPX's and a SPARC 10 workstation as part of plans to teach CIM using multiple platforms. The DOS-based environment is hierarchically controlled using a graphical flowcharting software package that was developed in-house. This approach will be described in detail in section E, and constitutes the focus of this paper.

B. CIM Control Hierarchy:

Components

At the heart of any computer control hierarchy are a few key components. A *network* (or several interconnect local area networks or LANs) provide data pathways and electronic data exchange (EDX) capabilities to the manufacturing system. A common *database* is the central repository for knowledge stored in and available across the network. Both of these are requisite but can exist at various levels of technological sophistication. For example, the database can be a complicated relational or object-oriented database, or as simple as information stored in a spreadsheet or as arrays in program. The network can be several LANs connected by a fiber optic network, or all controlled machines can be hardwired to a central host within a short distance. The idea here is that the concept is very different from the realizations of the concept, and can be very diverse.

One communication model developed and proposed by the International Standards Organization (ISO) is the Open System Interconnect (OSI) model [6,7]. This is actually a set of standards developed in conjunction with IEEE. While not all systems will implement all layers in the model, it is valuable for vendors of system components to provide equipment that can be used in a wide variety of systems. The model is shown in Table 1.

Table 1. OSI Communication Model

Layer	Name	Function
7	Application	Functional connectivity between application and network
6	Presentation	Message translation into receiver-readable format
5	Session	Arranges conversation between sender and receiver
4	Transport	Message integrity from sender to receiver
3	Network	Routing of data among network nodes
2	Data Link	Transmission as packets
1	Physical	Defines attachment (cabling) requirements

It may be noted that this model allows for flexibility among different types of hardware and operating systems.

Levels of Control

Consistent with the NIST-AMRF hierarchy, several basic levels of control must exist for a successful CIM implementation:

1. Area (supervisory)
2. Workcell (function)
3. Machine (task)

Note that a one-to-many relationship exists between Workcell and Machine, and a one-to-many relationship exists between Area and Workcell, with many levels of Area's possible, each with a one-to-many relationship with the next higher level. An example of an area might be Department; another example of an Area would be a Factory Shop Floor, obviously composed of many Departments. This allows a recursive relationship to be defined among the various levels.

Different strategies exist for division of labor and control among these levels. Two common denominators exist: The scheduling horizon for tasks decreases when comparing Area to Workcell, and task granularity (or specificity) increases when comparing Area to Machine (or lower level Area's)

C. Equipment Control Requirements:

The components controlled tend to be NC machine tools or robots, but can be any machine or device, such as an automated guided vehicle (AGV) where the behavior can be controlled (in a discrete or continuous fashion). Here the term *control* can be defined as any modification of behavior that is not originated by the device whose behavior is modified.

Functionality

A certain level of basic functionality must exist for a device to be controlled in any real sense on the manufacturing floor. First, the machine must be *reconfigurable* (flexible; modular), for it to perform in different capacities to produce various components. This defines the *physical* configuration of the equipment.

The *logical* configuration of the equipment refers to the sequence of programmed tasks the device performs. Therefore, the device must be *reprogrammable*. To this end, the ability of the machine to upload and download programs is essential. This is consistent with the requirements placed on a system using supervisory computer control. The devices being controlled must be able to report on their status at all levels, including tool status (broken, tool offsets) and program status (done, halt, pending, error). The devices must also be able to receive and respond to external messages across the communication network, such as Start, Stop, and respond to queries.

The supervisory levels of control (Workcell and Area) must be able to schedule tasks for the next lower level of control; control deadlock; selectively start and stop machines in sequence or parallel; upload and download programs selectively as needed; monitor tasks; and report to the next higher level of control.

D. Fundamentals of CIM Control Software:

Based on the AMRF CIM Hierarchical Control and portion of the OSI communication model, CIM software can be written generically and customized through device drivers and programs. Each level in the AMRF model would be controlled by a computer (a CPU). The features of a CIM control software have to at least satisfy the requirements in section C. Proper communication is the key issue and its capability for allowing flexibility and modularity is of utmost importance due to the concurrency and agility of modern engineering. Four features in a CIM Control Software stand out as the backbone of the algorithm: communication, system monitoring with feedback, logic condition check and manageability of the system.

Communication must not only allow for a central supervisory control unit to issue commands to the cell CPU control unit but also to receive and process feedback information as to the status of the cell cpu and consequently the lower level equipment. If, for example, the NC machine tool failed, a sensor signal should halt NC machining instantaneously and a message goes to the cell cpu. Either the cell cpu takes corrective action or it gets passed to the supervisor cpu for action. In either case, this breakdown will most likely affect the other machinery or cells in the CIM environment. The supervisor cpu has to balance the activity of other cells such that queue sizes of workstations are maximized and yet bottlenecks are not created. The workstations with long process times should take this opportunity to catch up with production. Others may eventually have to be shut down until the problem is solved. This type of logical control can only happen when there is absolute flexibility in the entire system. This is the concept that is being pursued and taught to the students.

To have true *monitoring capability*, each variable in a workstation, consequently in a cell must be represented by a parameter whose values can be updated, monitored and changed. Messages can be passed back and forth. For example, in the above tool breakage, a simple monitoring parameter is to record the operating status of the NC machine. If under any circumstance the machine breaks down due to machine error or process planning error, this status parameter will be changed by the cell cpu. Depending on the nature of the action and the hierarchical level at which it is taken, this status parameter value may be changed again, perhaps many times, until its value is back to a 'GO' representation and the cell cpu will restart the NC machine.

Logical Control allows the system to have some intelligence in decision making. Most of the time, this logic only involves checking *when* the workstations are free or idle waiting for work. As an example, if a material handling system such as an automated guided vehicle (AGV) has to deliver a tray of parts to an assembly workstation, the workstation must be free to receive it and that there is an available tray to be picked up. The cell cpu for this AGV will sense the value of a parameter for availability, the location of the tray, the destination and its occupation status. When all these parameters are satisfied, then the AGV moves, selects a shortest available path, picks up and delivers. In a more intelligent system, it can predict the length of time it takes for the vehicle to travel from point A to point B and then *pre*-arranges for the pick up so that the assembly workstation will not have idle time. This type of system takes a very comprehensive algorithm and development of such a system is beyond the scope of this presentation. It requires research into, for example, use of sophisticated scheduling and queuing theory, fuzzy logic theory to accommodate such intelligence.

When software is easy to use, to learn, to setup, to implement and to improve, there is promise in its *manageability* and wide-spread application. Aside from setting up a computer network (e.g. ethernet with a MAP type protocol), modules in the software should guide the user to generate device drivers which are the key to controlling equipment with feedback signals. In implementing the monitoring concept and the logic condition checks, a visual process is necessary for the user to create and grasp the true control of a very complicated production system. Since there is much logic flow to be tracked throughout the environment, the best representation for generating such a CIM control system is through the use of *flow charts*. Therefore, flowcharting is an ideal method for the user to visualize and create a customized CIM Environment control application.

E. Application at GMI

The CIM philosophy at GMI leans heavily towards the hierarchical control of manufacturing environments based on the AMRF model. Three levels (supervisor, shop-floor and equipment) are used. The software was written by a group of students led by Wes Faler, and is now commercially available through CIMULUS, Inc.

For illustrative purposes, a simple example will be used throughout this discussion. A rocket of four parts (nose, body, left and right fins) is to be assembled (see Figure 3). The parts arrive as an assembly kit of the four separate parts via an automated guided vehicle into an AGV stand where the tray will be pulled into a common workarea for two robots arranged in close proximity. The body is to be fitted into the nose cone. The left/right fins are to be inserted into the body. Due to stability and tolerancing issues during assembly, the nose cone has to be placed on the fixture (stand) prior to the other assembly steps. Two workcells are working cooperatively, each is controlled by a cell PC, MORIA and LOTHLORI. MORIA controls robot KILI while LOTHLORI controls BOMBUR. Collectively they form a workcenter. The supervisory PC is named GANDALF. Figure 4 shows the hierarchy of control. The assembly sequence is as follows: (1) BOMBUR pushes the tray into the workarea, (2) BOMBUR places nose in stand while KILI installs the left/right fins, (3) BOMBUR places body (with fins) on the nose while KILI goes to HOME position, and (4) BOMBUR pulls tray out of assembly area into the AGV stand.

Robot program names are:

BOMBUR	PUSHTRAY	/* push tray to work area
BOMBUR	PUTNOSE	/* place nose on stand
KILI	DOLFIN	/* install left fin
KILI	DORFIN	/* install right fin
BOMBUR	PUSHBODY	/* place subassembly to nose
BOMBUR	PULLTRAY	/* pull tray to AGV stand

Flowchart names are:

GANDALF	ROCKET	/* for supervisor
MORIA	ROCKETK	/* for MORIA controlling KILI
LOTHLORI	ROCKETB	/* for LOTHLORI controlling BOMBUR

The key for communication among the supervisor PC and cell PC's is a phenomenon called "message passing". Each message constitutes a device or network node I.D., an action code and an object code. (e.g. BOMBUR execute PUSHTRAY). These messages can be represented in control blocks, each with an INPUT and an OUTPUT. Figure 5 shows the block structures. If two robots are within one workcell, single or simultaneous messages can be issued for sequential and parallel operations respectively. For example if BOMBUR and KILI were both controlled by MORIA, then parallel operations could be as shown in Figure 6.

The handshaking process provides monitoring capability for supervisory control through the use of global parameters whose value gets passed throughout the network system, accessible to any node PC. In figure 7, handshaking is provided by a global parameter $S which tracks the status of robots in the cell by the supervisory PC. For example, when $S=0 the cell is idle, $S=1 the cell is busy, $S=3 the cell has completed the job, $S=99 the cell has problems. The supervisor waits for $S=3 or 99 before further action is taken.

To visualize the above message passing, handshaking and control processes, let us examine the flowcharts for the example of the rocket. Appendix A shows a set of symbols. Appendix B gives the complete set of flowcharts for GANDALF, MORIA, and LOTHLORI. Assume that the CIM system equipment are initialized and ready to operate. GANDALF starts flowchart ROCKET. It first sets $POWER=0 to ensure the cells do not operate since the supervisor is not ready. Then it starts up MORIA's ROCKETK and LOTHLORI's ROCKETB before interfacing with the user through UIF (User InterFace) module to get input $N (number of parts to make) and $NB (number of parts per batch). Next it initializes other variables. NOTE that $POWER=1 is a signal to MORIA and LOTHLORI to start processing. While $Parts < $N, the cells keep operating. When the required $N parts are produced, the flowchart assigns $POWER=0 for cell PC's to shut down. MORIA and LOTHLORI both wait for the signal $POWER=1 from GANDALF to begin and each produces a batch at a time. The cell PC's cross check status signals, namely, MORIA checks for status of BOMBUR ($B) and LOTHLORI checks for status of KILI ($K) while setting their own status parameters $K and $B respectively. On comparison of the two cell PC flowcharts, PUSHTRAY is a sequential process, while PUTNOSE and DOLFIN/DORFIN are parallel operations since LOTHLORI waits for KILI to be done ($K=9) with DOLFIN and DORFIN, while MORIA waits for BOMBUR to finish with PUTNOSE ($B=9). Similarly, PUSHBODY/PULLTRAY and HOME are parallel operations. When assembly is completed, the parameter values are updated, messages are printed on the screen through UIF and the procedure continues until the supervisor stops the production.

E. *Student Project:*

To make the concepts of hierarchical control meaningful to the students, aside from lab exercises, the entire class is assigned one project where the 30-50 students will design, plan and produce one product of their choice. Groups of 5-8 will work on one aspect of the project from design to finish. For example, if a desk set with a base and a cover and pockets was to be made, the possible separate projects could be: (1) design and manufacture of the base, (2) design and manufacture of the cover, (3) design the layout and delivery of raw material, (4) assembly of the desk set, (5) packaging and storage of the assembly and (6) material handling and transfer system configuration and routing. These separate projects will be integrated with the hierarchical control flowcharting system. The students learn to communicate, interact, adhere to standards and pass product information throughout the duration of the project. Comments from the students include "out of the entire curriculum, this is the most practice-oriented course".

Figure 3: Rocket Assembly

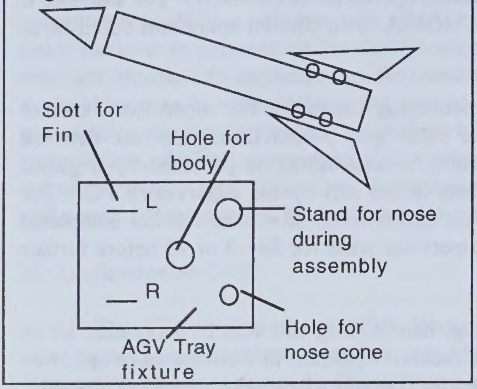

Slot for Fin
Hole for body
L
Stand for nose during assembly
R
Hole for nose cone
AGV Tray fixture

Figure 4: Example CIM Hierarchy

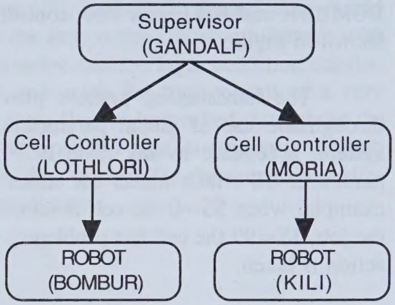

Supervisor (GANDALF)
Cell Controller (LOTHLORI)
Cell Controller (MORIA)
ROBOT (BOMBUR)
ROBOT (KILI)

Figure 5: Blocks Concept

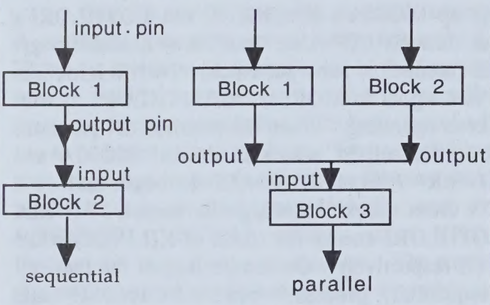
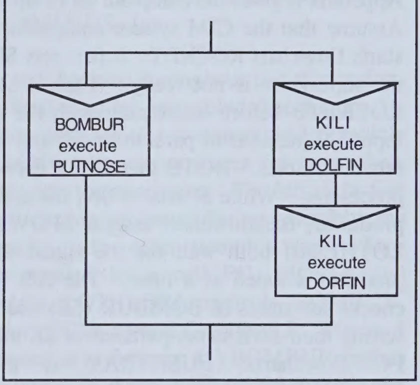

input · pin
Block 1
output pin
input
Block 2
sequential
Block 1
Block 2
output
output
input
Block 3
parallel

Figure 6: Parallel Structures


BOMBUR execute PUTNOSE
KILI execute DOLFIN
KILI execute DORFIN

F. References:

[1] Shrensker, W.L. "CIM Computer Integrated Manufacturing: A Working Definition", SME Blue Book Series, December 1990.

[2] Jones, A.T., and C.R. McLean, "A Proposed Hierarchical Control Model for Automation Manufacturing Systems", J. Manufacturing Systems, pp. 15-25, Vol. 5, No. 1, 1986.

[3] Koska, D.F., and J.D. Romano, "Countdown to the Future: The Manufacturing Engineer in the 21st Century", A.T. Kearney Research Study, Profile 21, Executive Summary, SME, Fall 1988.

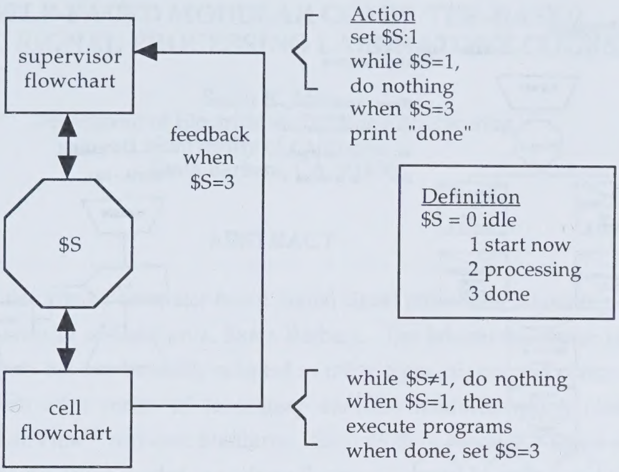
[4] Arthur, A.A., D. Wells, and P.J. Demers, **Curricula 2000 Workshop Proceedings: Ideal Models in Manufacturing Education**, SME, February 1990.

[5] Veilleux, R.F., and L.W., Petro. **Tool and Manufacturing Engineers Handbook, Volume V: Manufacturing Management**, 4th Edition, Dearborn, Michigan: Society of Manufacturing Engineers, 1991.

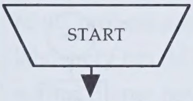
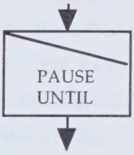
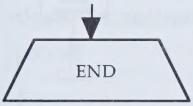
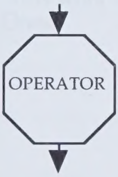
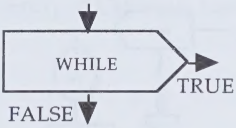
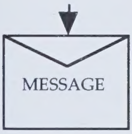
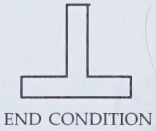
[6] Bedworth, D.A., M.R. Henderson, and P.M. Wolfe, **Computer-Integrated Design and Manufacturing**, McGraw-Hill, 1991.

[7] Mitchell, F.H., Jr., **CIM Systems**, Prentice Hall, 1991

Figure 7: Handshake

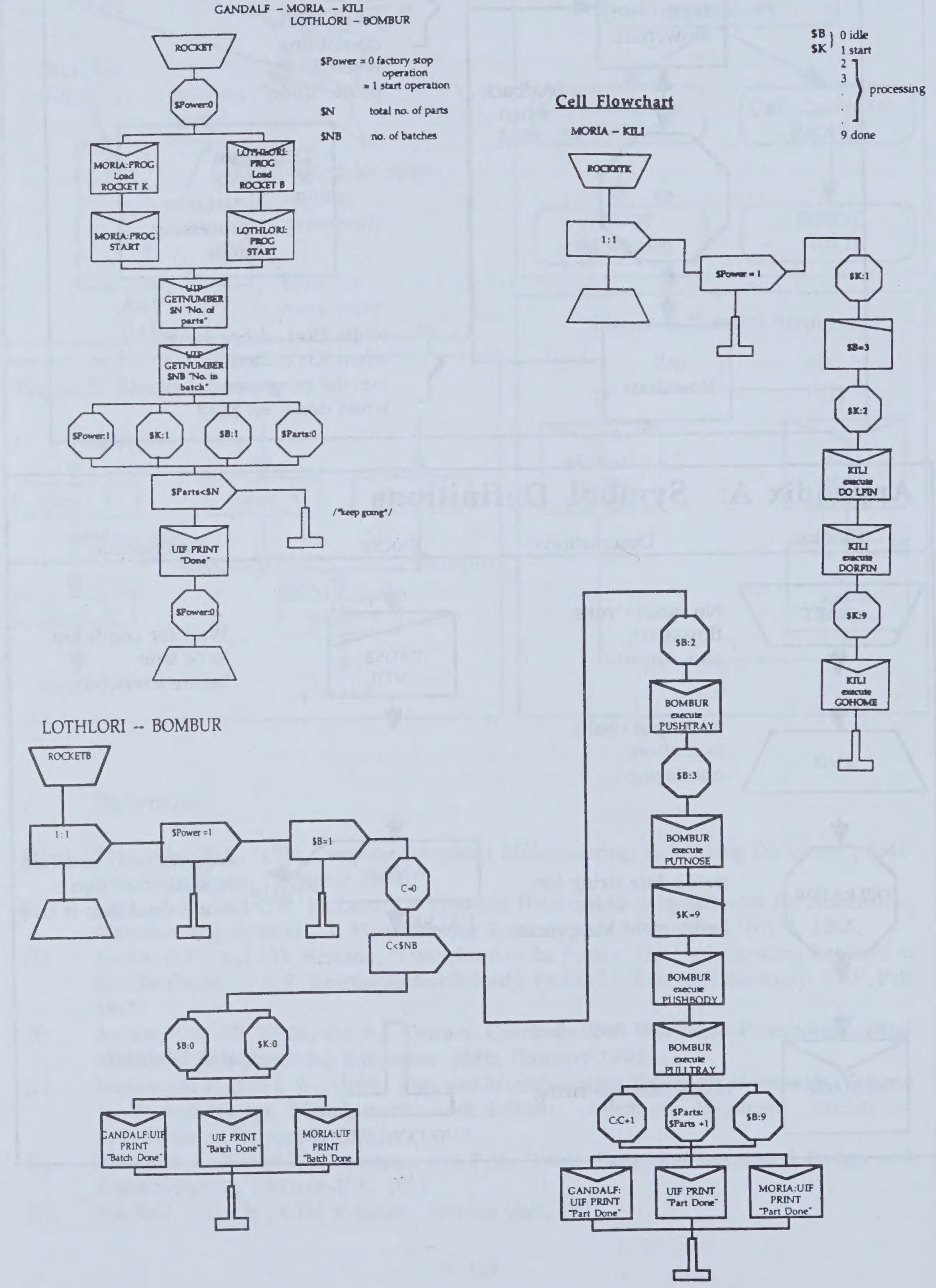

supervisor flowchart
feedback when $S=3
$S
cell flowchart
Action
set $S:1
while $S=1,
do nothing
when $S=3
print "done"
Definition
$S = 0 idle
1 start now
2 processing
3 done
while $S≠1, do nothing
when $S=1, then
execute programs
when done, set $S=3

Appendix A: Symbol Definitions

Blocks	Descriptions	Blocks	Descriptions
	No input - runs flowchart, one output	 PAUSE UNTIL	Wait for condition to be true before execution
	no output - halts execution, one input		
	needs data string for setting variables, assignment statements	 WHILE TRUE FALSE	requires test condition repeats operation while condition is true
	needs message string	 END CONDITION	end condition

Appendix B:

Supervisory Flowchart


GANDALF – MORIA – KILI
LOTHLORI – BOMBUR
ROCKET
$Power:0
$Power = 0 factory stop operation
= 1 start operation
$N total no. of parts
$NB no. of batches
MORIA:PROG Load ROCKET K
LOTHLORI: PROG Load ROCKET B
MORIA:PROG START
LOTHLORI: PROG START
UIF GETNUMBER: $N "No. of parts"
UIF GETNUMBER $NB "No. in batch"
$Power:1
$K:1
$B:1
$Parts:0
$Parts<$N
/"keep going"/
UIF PRINT "Done"
$Power:0
Cell Flowchart
MORIA – KILI
$B = track BOMBUR
$K = track KILI
$B $K 0 idle
1 start
2
3
.
.
.
processing
9 done
ROCKETK
1:1
$Power = 1
$K:1
$B=3
$K:2
KILI execute DO LFIN
KILI execute DORFIN
$K:9
KILI execute GOHOME
LOTHLORI -- BOMBUR
ROCKETB
1:1
$Power =1
$B=1
C=0
C<$NB
$B:0
$K:0
GANDALF:UIF PRINT "Batch Done"
UIF PRINT "Batch Done"
MORIA:UIF PRINT "Batch Done"
$B:2
BOMBUR execute PUSHTRAY
$B:3
BOMBUR execute PUTNOSE
$K=9
BOMBUR execute PUSHBODY
BOMBUR execute PULLTRAY
C:C+1
$Parts: $Parts +1
$B:9
GANDALF: UIF PRINT "Part Done"
UIF PRINT "Part Done"
MORIA:UIF PRINT "Part Done"

A SELF-PACED MODULAR COMPUTER-BASED DIGITAL SIGNAL PROCESSING LABORATORY COURSE †

Sanjit K. Mitra
Department of Electrical & Computer Engineering
University of California
Santa Barbara, CA 93106

ABSTRACT

This paper describes a computer-based digital signal processing laboratory course developed at the University of California, Santa Barbara. The laboratory course is based on Macintosh computers but can be easily adapted to other types of personal computers. The courseware consists of a series of laboratory exercise modules which make use of MATLAB® and LabView® software platforms. Each module contains a series of projects containing the basic programs needed to perform the experiments. In order to ensure that the student understands the basic principles of the topic under investigation or the key features of the program being used, each project description is followed with a series of questions which the student need to answer before moving on to the next project. A skeleton laboratory report is provided for each exercise module. The student performing the exercises includes results of the projects and answers to the questions as he/she progresses through the exercises. The laboratory has 24 hours access permitting the student to work on the exercises outside the normal 3 hour period assigned per week. Each weekly exercise contains extra projects designed for motivated students to work beyond what average students can complete in a 3 hour period.

---

† This work was supported in part by a University of California Instructional Improvement Grant.

Workstations in a Communication Systems Laboratory

Bruce A. Black
Department of Electrical and Computer Engineering
Rose-Hulman Institute of Technology
Terre Haute, IN 47803

Abstract

In the spring of 1992 a new electrical engineering laboratory built around the Hewlett-Packard Visual Engineering Environment (VEE Test) was installed in the Advanced Circuits Laboratory at Rose-Hulman Institute of Technology. At each laboratory station an IEEE-488 bus connects an oscilloscope, a function generator, a multimeter, and a programmable power supply to an HP 9000/375 workstation. This paper describes our experience in using the Advanced Circuits Laboratory for projects in the course Introduction to Communication Systems. Students use VEE Test to acquire and display frequency response and transient response data that formerly were collected by hand. The students use Matlab for data manipulation and analysis, for calculation, for digital filtering, and for computation of Fourier transforms. It has been found that the time saved by automating the more tedious aspects of data acquisition more than make up for time required by students to learn to use the workstation software. The ease with which calculations can be performed and results compared allow students to achieve a greater understanding of the relationship between measured and predicted results. Finally, the ease with which data from a variety of sources can be read into Matlab and graphed together improves the quality of students' written reports.

Introduction

In the spring of 1992 a new electrical engineering laboratory built around the Hewlett-Packard Visual Engineering Environment (VEE Test) was installed in the Advanced Circuits Laboratory at Rose-Hulman Institute of Technology. The goal of the Advanced Circuits Laboratory is to encourage conceptual understanding, open-ended problem-solving, and engineering design by providing tools that help students with the repetitive and/or tedious parts of data collection and manipulation.

Laboratory equipment includes the following items:

Quantity	Description
11	Hewlett-Packard HP 9000/375 Workstation (including 1 server)
10	Hewlett-Packard HP 54504A Digitizing Oscilloscope
10	Hewlett-Packard HP 8116A 50 MHz Pulse/Function Generator
10	Hewlett-Packard HP 34401A Multimeter
10	Hewlett-Packard HP 6632A Power Supply
1	Hewlett-Packard Laserjet IIIsi Printer

At each laboratory station an IEEE-488 bus connects the oscilloscope, function generator, multimeter, and programmable power supply to the workstation. The workstations are networked to form a "cluster" that is connected to the campus network, and ultimately to the Internet. The Laserjet IIIsi printer is connected to the server, but can be accessed from each workstation.

Laboratory software includes:

Quantity	Description
11	Hewlett-Packard Visual User Environment (VUE) (A graphical user interface.)
10	Hewlett-Packard Visual Engineering Environment (VEE Test)
50	Matlab 3.5 with Signal Processing, System Identification, Optimization, Control, and Robust Control Toolboxes
50	Simulab
10	Wingz spreadsheet
10	Mathematica 2.2

VEE Test provides a software environment in which students (and faculty) build programs for data acquisition, analysis, and display by using predefined visual objects representing signal generators, FFT routines, multimeters, strip charts, and XY plots. Icons representing these objects are manipulated on the workstation screen and connected using "wires" instead of lines of written code. The objects that represent test instruments utilize the IEEE-488 bus to control and collect data from the actual instruments. Matlab, Mathematica, and Wingz provide powerful tools for analyzing data, for signal processing, and for comparing predicted and measured results.

Unlike other laboratories where only one or two workstations with IEEE-488-based instrumentation are available, the Advanced Circuits Laboratory attempts to provide routine access to computer workstations for all students during the course of a laboratory. Workstations are connected to the campus-wide network so that after a scheduled laboratory session students can access their data from any workstation on campus.

The Advanced Circuits Laboratory is shared among a range of electrical engineering courses offered at the Institute. Courses which have used the laboratory include Circuits and Systems, Electronic Circuits, Introduction to Logic Design, Nonlinear Electronics, Traveling Waves, Instrumentation, and several courses in Communication Systems.

This paper describes our experience in using the Advanced Circuits Laboratory for five of the nine laboratory projects in the required junior-level course EE391, Introduction to Communication Systems. The laboratory projects developed for this course include

- Frequency Response of a Filter
- Pulse Approximation of Signals
- Impulse Response and Convolution
- Filter Effects upon Periodic Signals
- The Discrete Fourier Transform.

The remaining four laboratory projects involve use of the spectrum analyzer and are performed in a different facility. In performing the experiments listed above, students use VEE Test to acquire and display frequency response and transient response data that formerly were collected by hand. The students use Matlab for data manipulation and analysis, for calculation, for digital filtering, and for computation of Fourier transforms. We have found that the time saved by automating the more tedious aspects of data acquisition more than makes up for time required by students to learn to use the workstation software.

The ease with which calculations can be performed and results compared allows the students to achieve a greater understanding of the relationship between measured and predicted results. Finally, the ease with which data from a variety of sources can be read into Matlab and graphed together improves the quality of the students' written reports.

Communication Systems at Rose-Hulman

The communication systems course sequence at Rose-Hulman comprises three quarter-length courses normally taken by electrical engineering students during their junior year. Students starting this sequence have completed a full year of Circuits and Systems and mathematics through differential equations. The first course, Introduction to Communication Systems, begins with material that might be found in a course on signals and systems. It's principal theme is Fourier analysis, but it also includes applications to filtering, sampling, and modulation. The second course, Communication Systems I, is a traditional treatment of modulation methods. The third course, Communication Systems II, introduces noise in analog and digital communication systems.

Introduction to Communication Systems is a required course for all electrical engineering and computer engineering majors. The course is normally offered in the Fall Quarter to an enrollment of about 90 students divided into three sections. Communication Systems I is required for electrical engineering majors only, while Communication Systems II is elective for all students. Each of these courses meets for lectures four times each week. In addition, Introduction to Communication Systems has an integral laboratory that meets weekly for three hours. As quarters at Rose-Hulman provide ten weeks of instruction, this allows time for between eight and ten laboratory projects. The first laboratory project, which for several years has been Frequency Response of a Filter, is assigned in the first week of the quarter, and builds on skills that the students have acquired in their circuits and systems courses during the previous year.

Frequency Response of a Filter

In this laboratory project students are asked to measure the insertion loss vs. frequency of a fifth-order butterworth filter. Students are asked to graph insertion loss vs. frequency both with and without a 50 Ω termination present on the filter. They are asked to use SPICE to analyze a model of the filter, and to account for differences between the measured and SPICE-predicted insertion-loss plots.

To perform the insertion loss measurements, students use the HP 8116A function generator and HP 34401 multimeter. Students are provided with a VEE Test "program" to control the function generator and multimeter and to produce the insertion-loss graph. The VEE Test "program" also allows the measurement results to be captured in a file that can be later read in by Matlab to allow plotting. Figure 1 below shows the VEE Test "panel" as provided to the students. Present on the "panel" are a start button and a graph. The VEE Test "program" is shown in Figure 2. Icons representing the function generator and multimeter are visible, as are icons controlling the selection of frequencies and output to a file. The "formula" icon contains the expression for calculating insertion loss in decibels from measured load voltage. An example of the graphical hard copy that VEE Test provides to the user is shown in Figure 3.

The frequency response laboratory project has several objectives. First, it provides a

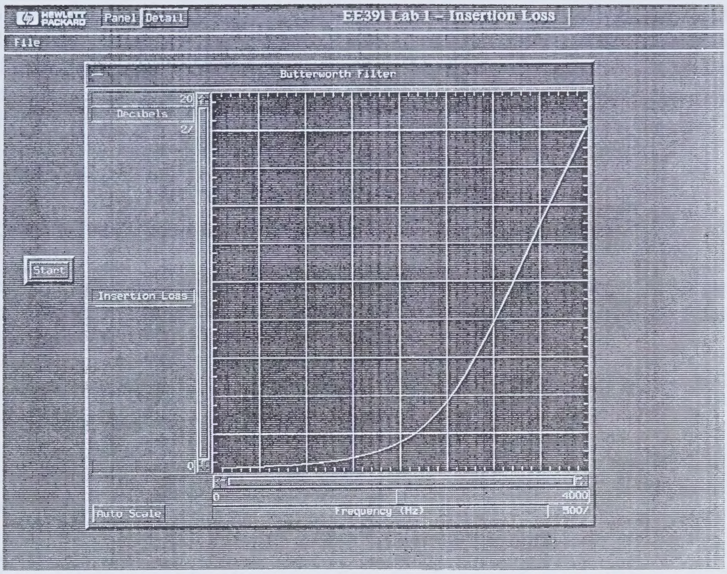

HEWLETT PACKARD
Panel
Detail
EE391 Lab 1 – Insertion Loss
File
Butterworth Filter
20
Decibels
2/
Start
Insertion Loss
0
0
4000
Auto Scale
Frequency (Hz)
500/

Figure 1 VEE Test Panel

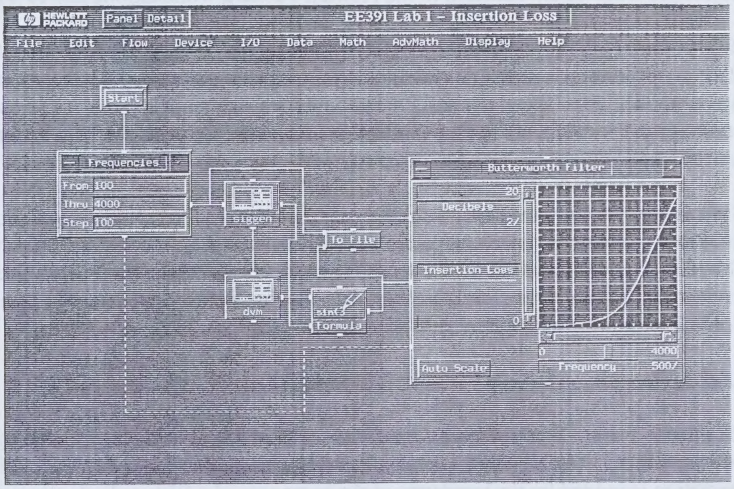

HEWLETT PACKARD
Panel
Detail
EE391 Lab 1 – Insertion Loss
File
Edit
Flow
Device
I/O
Data
Math
AdvMath
Display
Help
Start
Frequencies
From 100
Thru 4000
Step 100
siggen
To File
dvm
sin()
Formula
Butterworth Filter
20
Decibels
2/
Insertion Loss
0
0
4000
Auto Scale
Frequency
500/

Figure 2 VEE Test Program for Measuring Insertion Loss

review and a link to Circuits and Systems, in which one of the final laboratory projects was a frequency response measurement done in the traditional way by hand. Second, it introduces several filtering concepts that will be pursued later in the course, such as insertion loss, cutoff frequency, and proper terminating impedance. Third, the lab allows comparison of

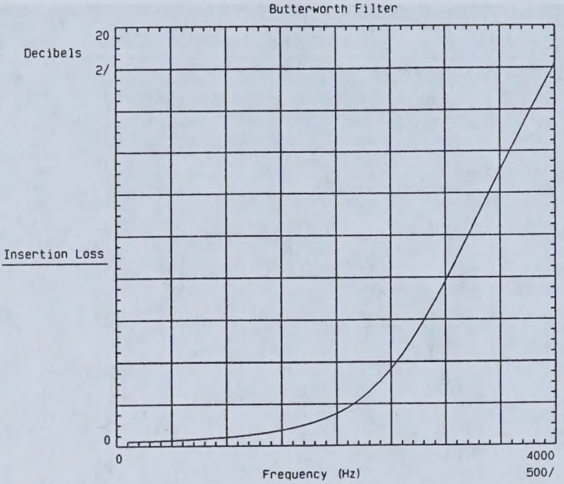


Figure 3 Graphical Output from VEE Test

performance of a model and an actual circuit, and challenges the student to try adjusting the parameters of the model to provide a good fit to the measured frequency response. Finally, this project provides an introduction to the Hewlett Packard workstations and to VEE Test.

In this initial lab project, students connect the physical filter to the instrumentation, but VEE Test is operated as a turn-key system that performs an automated frequency response measurement. The time saved in taking data allows time for students to learn to use the workstations, and for students to take multiple runs of data, for example with different terminating impedances.

Capturing measured insertion-loss data in a file allows the graphical capability of the computer to be used to enhance comparison of SPICE and measured results. Figures 4 and 5 show Matlab-generated plots of SPICE-predicted and measured insertion loss. The effect of the terminating impedance is clearly visible, as is the discrepancy between the measured insertion loss and that predicted by SPICE.

Pulse Approximation of Signals

In this lab project students investigate how a signal might be approximated by a staircase waveform. The signal in question is a single triangular pulse and the quality of the approximation is judged by the similarity of the responses of a fifth-order butterworth filter to the triangular and staircase inputs. This project introduces the concept of impulse response, and provides motivation for the development of the convolution integral, which is being discussed in the lectures at about the time this project is performed. By focussing attention on the filter output we lead the students to discover that a fairly coarse staircase approximation can be an accurate representation of a waveform in some applications. The approximation issue comes up again when Fourier series is presented later in the course.

This lab project is carried out entirely using Matlab. The fifth-order butterworth filter is available from the Matlab Signal Processing Toolbox. This project will normally be the students' first introduction to Matlab, and so a significant part of the three-hour lab session is spent in providing Matlab instruction. The actual lab project is of only moderate complexity and students can complete it outside of scheduled laboratory hours if necessary. The Matlab skills that are developed are used again in this and in several following courses in communication systems and controls.

Impulse Response and Convolution

In the laboratory the students construct a simple low-pass RC circuit. They then use three methods to determine the response of this circuit to a five-volt pulse of one time-constant duration.

- The HP 8116A function generator is used to generate the required pulse. The response is observed using the HP 54504A digital oscilloscope.

- The impulse response of the circuit is calculated. The convolution integral is used to calculate the response to a pulse.

- SPICE is used to model the circuit and obtain a pulse response.

To interpret the results of this lab exercise it is necessary for the students to compare the three pulse responses and to verify that they are sufficiently similar. In past years considerable time was spent sketching waveforms from oscilloscope displays with indifferent results. Now VEE Test is used to capture the waveform displayed on the digital

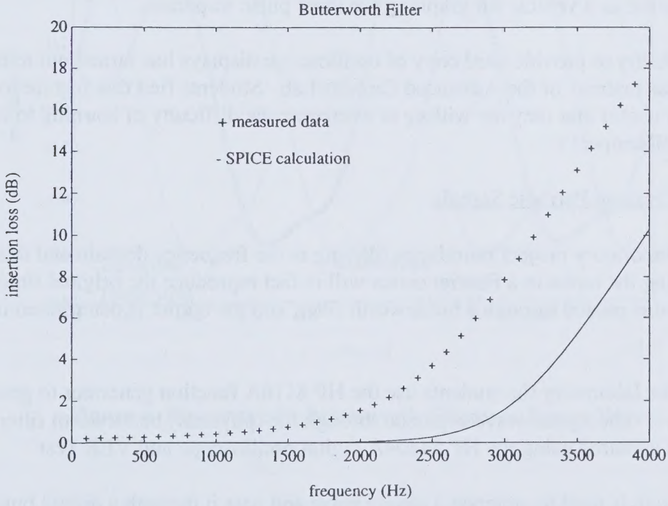


Figure 4 Measured and Calculated Insertion Loss

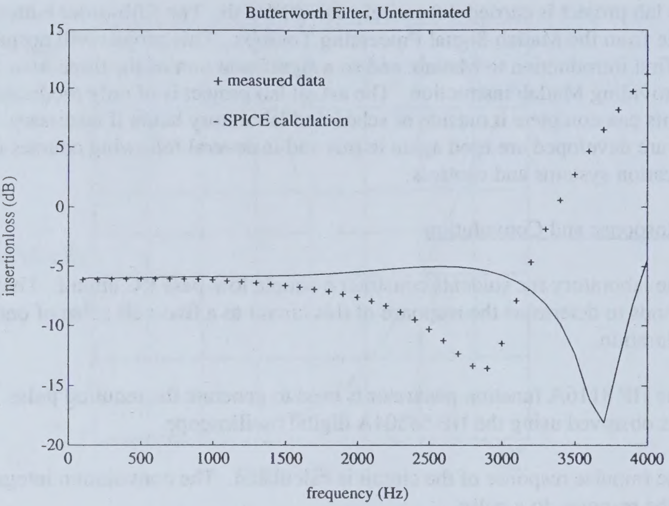


Figure 5 Measured and Calculated Insertion Loss, Unterminated Filter

oscilloscope. It turns out to be surprisingly simple to transfer a display from the oscilloscope to the workstation display and to obtain a printout of the result. It requires only slightly more effort to transfer the oscilloscope display to a file that can be read by Matlab. This allows Matlab to serve as a vehicle for graphing the three pulse responses.

The ability to provide hard copy of oscilloscope displays has turned out to be one of the most popular features of the Advanced Circuits Lab. Students find this feature to be sufficiently useful that they are willing to overcome the difficulty of learning to use the digital oscilloscopes!

Filter Effects upon Periodic Signals

This laboratory project introduces filtering in the frequency domain and demonstrates that summing the terms in a Fourier series will in fact reproduce the original time signal. A square wave is passed through a butterworth filter, and the output is determined in three ways:

- In the laboratory the students use the HP 8116A function generator to generate a square wave. The square wave is passed through the (physical) butterworth filter, and the response is captured using the HP 54504A digital oscilloscope and VEE Test.

- Matlab is used to generate a square wave and pass it through a digital butterworth filter as in the Pulse Approximation of Signals project.

- The third method of generating a square-wave response is a little more intricate and involves several steps: First students use the pole-zero diagram for the butterworth filter transfer function to develop an analytic expression for the frequency response. Next, they are asked to evaluate this frequency response at the first nine harmonic frequencies of the square wave. The Fourier series coefficients of the input square wave are then multiplied by the frequency response values to develop Fourier series coefficients for the filter output. Finally Matlab is used to generate sinusoidal waveforms having amplitudes and phases determined by the output Fourier series. These sinusoids are added to produce a predicted filter response.

Evaluating the frequency response of a fifth-order filter at several frequencies is an overwhelming task without some kind of computational assistance. Students make very effective use of Mathematica, Matlab, or several varieties of scientific calculator to perform the calculations. As the calculated results are expected to agree with measured data, students are motivated to repeat the calculations until they are correct!

Figure 6 is a Matlab-generated graph showing the measured response to the butterworth filter, the response to the digital filter, and the response generated by summing the terms of the Fourier series. The three waveforms agree remarkably well if the phase is discounted. (The phase of the measured data depends on the oscilloscope trigger level, while the phase of the digital filter output is influenced by the filter delay, and the phase of the Fourier series is arbitrary!)

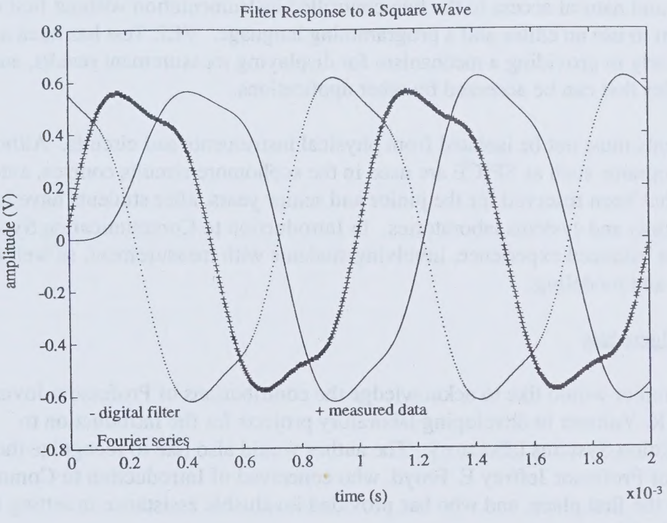


Figure 6 Response of a Butterworth Filter to a Square Wave

The Discrete Fourier Transform

A ten week quarter does not provide sufficient time for development of both continuous-time and discrete-time Fourier analysis. This lab project provides an "experimental" introduction to the discrete Fourier transform. In this project students use the fast Fourier transform capability of Matlab to investigate the transforms of several signals. The waveforms chosen: "impulses," pulses, sinusoids, and pulsed sinusoids, explore both the similarities and differences between the continuous and discrete cases. Where practical the students are asked to verify the Fourier transforms by hand calculation. Some of the lab time is spent discussing the significance of the results.

Conclusions and Challenges

Workstations and workstation-controlled equipment have the potential to place impressive power in the hands of students, but several challenges must be met to guarantee a favorable outcome. First, the laboratory must be *enabling*. Enhanced opportunities to learn can replace the tedium of data collection, but tedious data collection must not be replaced by tedious programming. Our approach is to introduce the computer gradually, and always in a context in which it will be perceived as useful. Ideally the investment in learning a new computer application package will be amortized over several courses. This is the case at Rose-Hulman with Matlab, which is used in several courses following Introduction to Communication Systems.

The laboratory must be *intriguing* to both faculty and students. Faculty and students alike are overwhelmed by the variety of computer tools and packages available today. One of our principal concerns in selecting software for the Advanced Circuits Laboratory was to allow easy and natural access to the bus-controlled instrumentation without first requiring the user to learn to use an editor and a programming language. VEE Test has been a success for us, particularly in providing a mechanism for displaying measurement results, and collecting data into files that can be accessed by other applications.

Students must not be isolated from physical instruments and circuits. Although circuit analysis programs such as SPICE are used in the sophomore circuits courses, automated data collection has been reserved for the junior and senior years, after students have had a full year of circuits and systems laboratories. In Introduction to Communication Systems we try to provide a balanced experience, involving students with measurement, as well as with simulation and modeling.

Acknowledgements

The author would like to acknowledge the contributions of Professors Jovan E. Lebaric and David R. Voltmer in developing laboratory projects for the Introduction to Communication Systems laboratory. The author would also like to recognize the invaluable assistance of Professor Jeffrey E. Froyd, who conceived of Introduction to Communication Systems in the first place, and who has provided invaluable assistance in setting up and operating the laboratory.

This paper is based on work supported by the National Science Foundation under Instrumentation and Laboratory Improvement Grant No. USE-8951436. Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the author and do not necessarily reflect the views of the National Science Foundation.

References

1. "HP Visual User Environment 3.0 User's Guide." Corvallis, OR: Hewlett-Packard Company, 1992.
2. "Using HP VEE-Engine and HP VEE-Test," edition 1. Corvallis, OR: Hewlett-Packard Company, 1991.
3. "PRO-MATLAB User's Guide." South Natick, MA: The MathWorks, Inc., 1991.
4. Wolfram, Stephen, "Mathematica: A System for Doing Mathematics by Computer," second edition. Redwood City, CA: Addison-Wesley Publishing Company, 1991.

TEACHING PRINCIPLES AND EXPERIMENTATIONS OF TELECOMMUNICATION NETWORKS

Joseph Y. Hui
Dept. of ECE, Rutgers University, Piscataway, NJ 08855-0909
jhui@winlab.rutgers.edu

Abstract

The teaching of networking is too often overloaded with standards and acronyms. We suggest how material for networking may be organized, with emphasis on the teaching of principles and experimentation. We describe courseware designed to instill a cohesive set of principles which are relatively technology independent. A graphical simulation and design tool, as well as protocol programming exercises are described which provide valuable experimental and hands-on experience. Also, students could prototype multi-media networking experiments on a switched ATM network.

`The laboratory is funded by NSF under grant NSF ILI 9250974`

1. Instruction of Networking - Status and Course Development

1.1 Current Curriculum Deficiencies

Instruction on switching, telephony [1-5], and integrated networking, a required component of most European EE curricula, is relatively neglected in the American curricula which focus primarily on computer communication [5-10]. This lopsided treatment results from the fact that prior to the divestiture of the Bell System, telephony and integrated public networking were taught almost exclusively within the Bell System by Bell Laboratories. With the divestiture and the technological revolution of fiber optics, integrated service networks, and computing technologies, academia should assume more responsibility in training engineers to build the future communications infrastructure.

The telecommunication industry is at least as large and multi-disciplinary as the computer industry. Unfortunately, curriculum development for communication networking has lagged substantially behind that for computing. While an undergraduate may major in computer engineering, he may not find an undergraduate course on networking offering the following desirable elements:

1. Teaching principles rather than an overdose of network acronyms and standards.
2. An integrated treatment of telephony, data, and video networks in lieu of separate courses.
3. Graphical computer tools for observing and illustrating network phenomena, which often cannot be properly explained by textbook or mathematical equations.
4. Hands-on experience with a real network and elementary network programming skills.
5. Team work on larger project, which is the predominant practice for networking.
6. Some treatment of optical networking rather just than low speed data communications.

These missing elements should be incorporated in a new arrangement of course material and a new laboratory setup. However, these elements can be pedagogically beneficial only if the larger question of purpose and framework of an undergraduate network course is properly defined.

1.2 A New Framework for Improved Instruction

The most fundamental question is: What should be taught and how should the material be structured? This is a difficult question for the *field* of networking which comprises diverse disciplines such as:

1. *Electrical Engineering*: access, multiplexing, transmission, switching, network design, etc.
2. *Computer Engineering*: algorithms, protocols, software, database, operating system, etc.
3. *Operations Research*: network optimization, stochastic performance analysis, etc.

The classical framework for undergraduate (and graduate) network courses has been predominantly the OSI 7 layer model, with textbook chapters describing the layers bottom-up [5-11]. This framework adopts the protocol view of predominantly data communications. While the OSI model is a desirable standard for communication, some networks such as telephone networks or high speed networks may not fit well with the framework. Also, the framework leaves student the impression that functionalities such as error correction, flow control, routing, information fragmentation and labeling are surgically defined for the layers. As the instructor proceeds to explain the details of a particular protocol suite within the OSI framework, he often finds functionalities repeated intricately at many ill-defined layers.

An alternative framework emerged from several years of experience in teaching networks. An analogy is drawn between the teaching of communication networks and life sciences, each involving the study of complex organizations or organisms. In studying an organism, one learns *Anatomy*, then *Physiology*, and finally *Clinical Measurement and Analysis*. In studying networks, a similar framework can be drawn, which is also loosely parallel to the previous EE/CE/OR dichotomy:

1. *Structural Analysis and Synthesis of Networks* -
   Network architectures for telephony, data network, and optical network. Circuit, packet, and optical multiplexers/switches. Network design, configuration, and routing.

2. *Functional Analysis and Synthesis of Protocols* -
   Protocol architectures. Protocol design, specification, and validation. Socket and network programming. Specific protocols for access, error and flow control, and connection setup.

3. *Analytical Study of Network and Protocol Performance* -
   Probability Theory. Traffic modeling. Elementary blocking and delay analysis.

1.3 Course Requirements and Support

An introductory course on networking is should give an overview and a road map for the field. Experimentation gives the students an intuitive rather than analytical grasp of this complicated field. Equally important to what is included in the instruction framework is what is excluded. The exclusions are covered in other subjects or used as follow-up special problems.

1. Modulation and transmission. Most undergraduates at the junior or senior level at Rutgers are well-familiar with digital electronics and communications, both required courses for ECE majors. The underlying physical layer can be briefly described as an unreliable digital pipe.

2. Application protocol and software. They are taught in two graduate software engineering courses.

3. Sophisticated mathematical optimization or stochastic analysis. The course emphasizes problem formulation, not solution techniques which are beyond the grasp of the average undergraduate. Solution techniques are embedded within the software developed and their use is taught.

As a junior or senior elective for both EE or CE options, the prerequisites are:

1. A course in Probability, Discrete Math, or Linear Algebra for enhanced math aptitude.

2. Knowledge of C, taught in a required programming course at Rutgers, is needed for protocol exercises.

2. Experimental Softwares and Equipments.

Experimentation is an important aspect of network instructions. The availability of hardware and software equipment for high speed and multimedia networking has opened new avenues for students to integrate their know-how through simulation and more importantly, prototyping protocols on real networks.

2.1 Laboratory equipments

Advances in computing and communication have generated fascinating possibilities for experimental network instruction.

1. The hardware revolution in workstation (RISC, scalable processor, graphics accelerator) has made the computationally intensive design, control, analysis, and simulation of networks and protocols affordable, accessible, and expeditious enough for undergraduate education for the first time.
2. The optical and integrated networking revolution shall dramatically increase the multi-media and information networking capability of these workstations. Recent developments in ATM and multimedia technologies make experimental instruction on real network experiments affordable.

Currently, our laboratory equipments consist of:

1. Eight SUN workstations, networked by an ethernet hub. Four of the workstations are equipped with multimedia hardware and software.
2. An ATM switch with 16 ports. Four of the workstations are currently connected via the host interface and fiber links to the ATM switch.
3. Modsim and Comnet software for simulation and analysis of communication networks. Based on the object-oriented Modsim software, we have produced a stable and user-friendly graphical package called **CANeT** (Computer-Aided Networking Tools).
4. Traffic measurement tools and protocol analyzers.

2.2 Software Development

Interactive graphical laboratory software is becoming necessary for network instruction because

1. Networks and protocols are very complex. Graphics facilitate their representation and visualization.
2. Network dynamics and complex network scenarios can be illustrated and studied most effectively through animation and color.
3. Network design, control, and simulation are computationally intensive tasks. The students should be able to use an extensive library of optimization and performance evaluation algorithms for real-time what-if analysis. Understanding such algorithms is beyond the grasp of the average undergraduate.
4. Solving a network problem involves analyzing the problem for its component problems and then synthesizing the solved component problems. Engineering iterates for this analysis-synthesis process is necessary before system objectives are achieved. Also, a team is the predominant work mode for solving complicated network problems. A well integrated software for solving component problems improves significantly the efficiency of this analysis-synthesis process and automates communication between engineers working in a team.

The software **CANeT** is designed for meeting these and several other objectives. Currently, it comprises more than 15,000 lines of code written using a high level Pascal-like object-oriented language called Modsim, which has extensive built-in graphics capability. The language and graphics environment can be learned and applied by undergraduates within weeks, and several undergraduates have contributed to the programming of **CANeT**. Modsim is preferred over C++ for its graphics capability, ease of learning, independence of machines and graphics interfaces, and ease of reading and modifying programs. The features of **CANeT** include:

1. A unified network graphical interface, employing extensively color, animation, and "oscilloscopes" for flexible, succinct, and cognitively friendly display of static and dynamic network parameters.
2. A 3 level menu system with potentially several thousand mouse-driven actions. The system has well-structured communication means between menu items at each level.
3. The top level menu consists of subjects in telecommunications such as network design, network and path management, network routing, flow control, packet switching, protocols, etc., as well as graphical editing for networks, traffic patterns, call definitions, etc.

4. Each subject in the top level menu inherits objects (network, node, link, path, etc.) and procedures (optimization algorithm, queueing analysis) from a common library. Since the same object in different subjects may have redefined attributes and communication means, the primary means of communications at the top is through standardized file structures.
5. System analysis tools is available for breaking a complex problem into a sequence of steps (subjects) at the top level menu. This facilitates easy iteration of the previously mentioned analysis-synthesis procedure for meeting system objectives.
6. The second level menu for each subject consists of action classes such as simulation, file storage and retrieval, measurement setup, post simulation analysis, major concerns for each subject, etc. The communication means at this level is primarily through TELL and ASK methods of objects.
7. The third level menu consists of actions or methods within each action class, such as routing methods for dynamic routing in networks, various means of display, clock and parameter setting for simulation, node and link placements, etc.

3 Experiment and Project Developments

Having described the lab setup, we now describe the experiments and projects to be facilitated by and conducted on the network.

3.1 Experimental Exercises

To illustrate the principles taught, we have to focus again on our previous framework concerning the *Structural, Functional, and Analytical* aspects of networking.

The primary principle to be taught in the *Structural* aspect of networking is how network resources are structured and utilized to make connections. Some concerns here are link and node topology, path and alternate routes configuration, and efficient multiplexing and switching. These structures must be designed and understood relative to system objectives such as cost, traffic demand, quality of service, service integration, and network reliability, etc. These are often conflicting objectives requiring engineering trade-offs. For example, link construction cost is a concave ∩ function of capacity due to the economy of scale, while delay versus traffic are convex ∪ functions. Also, risk of failure is a convex function of volume of traffic carried. These qualitative relationships are to be understood and traded-off through experimental investigation.

Several laboratory exercises have been developed. The first exercise uses our software tool to teach students circuit and packet switching. Students are required to make connections in multistage interconnection networks, through which they experience various blocking phenomena and learn how to rearrange circuits. The second lab, which is designed for studying dynamic routing for circuit switching, helps them to understand the dynamical and economical aspects of alternate routing. The third lab requires the student to design networks and analyze their reliability. These exercises illustrate the relationship among a few selected entities.

The primary principle to be taught for the *Functional or Procedural* aspects of networking is how protocol states are specified and distributed processes are created to achieve functionalities. For this purpose, students learn rudimentary network programming and test protocols on real networks. After introducing the OSI 7 layer protocol stack, TCP/IP is used as the experimental vehicle for achieving certain functionalities. TCP/IP is chosen due to its availability and its easily programmable socket utilities [11]. (Sockets constitute the interface between the transport and the session layers in the OSI framework. The CE students at Rutgers learn them in a required operating system course. Rudiments of sockets will be taught in our course.) Nevertheless, the scope of the functionalities has to be restricted significantly and a library of procedures built using sockets has to be constructed so that the student, given the time limitation, may not have to write a large program.

Protocols, implemented by the students, are tested over our ATM network in laboratory. We have devised protocol exercises implementing certain network functionalities:

1. *Traffic measurement for integrated networking.* Real or dummy traffic entities for multi-media are offered on the network. The students are required to program a probe for measuring traffic and delay in the network. Subsequently, the probe is used to study how information fragmentation affects performance.
2. *Reliable end-to-end packet delivery.* By using certain procedures of UNIX which drop, corrupt, or resequence packets, the students learn how to use timing and retransmission strategies for achieving reliable packet delivery. The effect of delay on such strategies is investigated.
3. *Multipoint and multi-media call management.* Reliable datagrams are used to establish virtual circuits for a 3 way call using 2 different media (perhaps data and images to start with). Issues of call management are examined.

The students may work in groups of two to three for prototyping these functionalities. Other subjects such as studying some aspects of flow control for high speed networks or media interaction in multi-media networks may be used as a special problem for the more ambitious students.

The primary principle to be taught for the *Analytical* aspect of networking is elementary teletraffic analysis. The qualitative relationships developed while studying the *Structural* aspects and the measurement experience developed while studying the *Functional* aspects are reinforced by more a priori probabilistic reasoning. Well-known relationships such as the Erlang-B formula and the delay formula for M/M/1 queues are derived and verified experimentally. The following experiments are devised:

1. *Statistical Advantages Achieved by Aggregating Traffic.* Using the **CANeT Call Definition** module, bursty and non-bursty calls can be defined and visually displayed. The student then simulates the aggregation of traffic and measures its mean and variance versus the degree of aggregation. The plots illustrate the benefits of the law of large numbers and the central limit theorem for high bandwidth transmission.

2. *Traffic Engineering using the Erlang-B or M/M/1 Delay Formulas.* These formulas usually express blocking/delay versus offered traffic for given capacity. An additional quantity of concern is carried traffic for blocking systems. An exercise is devised for the experimental measurements of these quantities. Depending on which parameter is measured, other quantities can be derived using a myriad of parameterized plots. The student should be able to explain the shape of the plots. The student can also see and measure graphically the statistics of blocked traffic.

3. *Queueing Network Performance.* The advanced students can use the **CANeT Queueing Network** or the COMNET software to simulate fast packet switching or protocol processors.

These exercises and others provide hands-on experience and indelible intuition for the three aspects of our instruction framework. We believe that pedagogically, these they are superior in explaining the basic concepts of communications, which until now are explained inadequately by either abstract equations or meticulously confusing industrial standards. Also, a field trip to a switching center (with lots of multiplexers, switches, and optical equipments) has always been a very positive educational experience.

3.2 Projects and Special Problems

Term projects and special problems could be extended versions of lab exercises. Samples of term projects include:

1. Develope software tools for network management.

2. Participate in the design and testing of a networked multimedia server. The performance bottlenecks are identified and improved upon.

3. Development of multimedia applications and study the communication quality of these applications.

These projects require the student to work in team and to learn how to manage larger projects. Such skills are invaluable when the student are employed in the telecommunication industries.

F. BIBLIOGRAPHY

1. J. Bellamy, *Digital Telephony,* John Wiley and Sons, New York, 1982.
2. M. J. Hills, *Telecommunications Switching Principles,* MIT Press, Cambridge and London, 1979.
3. J. McDonald, editor, *Fundamentals of Digital Switching,* New York: Plenum Press, 1983.
4. R. Rey, Editor, *Engineering and operations in the Bell System,* 2nd Ed., Bell Laboratories, 1983.
5. M.Schwartz,*Telecommunication Networks:Protocols,Modeling and Analysis,*Addison Wesley,1987
6. W. Stallings, *Data and Computer Communications*, 3rd Ed., MacMillan, 1991.
7. J. Hayes, *Modeling and Analysis of Computer Communications Networks,* Plenum Press, 1984.
8. D. P. Bertsekas and R. G. Gallager, *Data Networks*, Prentice Hall, Englewood Cliffs, 1987.
9. A. Tanenbaum, *Computer Networks*, 2nd Ed., Prentice Hall, 1988.
10. J. Walrand, *Communication Networks, A First Course*, Aksen Associates, 1991.
11. W. R. Stevens, *UNIX Network Programming*, Prentice Hall, 1990.
12. J. Hui, *Switching and Traffic Theory for Integrated Broadband Networks*, Kluwer, 1990.
13. C. Malamud, *Analyzing SUN Networks*, Van Nostrand Reinhold, 1991.

Upper Bound for the Capacity of a Hybrid Random Access System with Multi-User Coding

R.M. AL-Rumaih and Peter Mathys

Department of Electrical & Computer Engineering
University of Colorado
Boulder,Colorado 80309-0425

Abstract

In this paper we give an upper bound for the packet transmission rate (capacity) of hybrid random-access systems (HRAS) which use a combination of multi-user codes (MUC) with collision-resolution algorithms (CRA) to accomodate the bursty transmissions of many independent users on a single communication channel . We already have a lower bound for the capacity given in [7].

I. Introduction

We consider the problem of how a large number of users which generate bursty messages can share a common communication channel without central control. In the past, at least three different approaches have been taken for the analysis of such situations, namely the collision resolution (CR) approach, the multi-user information theory (MUIT) approach, and the spread spectrum (SS) approach [1]. Each of these approaches is usually pursued in virtual isolation from the others and each tries to model certain aspects of multi-user communication, while often grossly neglecting other important aspects.

We aim at combining the CR and the MUIT approaches by using a Hybrid Random-Access System (HRAS) which uses both collision resolution algorithms (CRA) and a multi-user code (MUC).

The system diagram is given in Figure 1. We assume that an infinite number of users transmit over a common channel. We use a T active out of N_c users MUC (with $T \leq N_c$) which is a set of N_c codes such that if any T or less users transmit simultaneously, the central receiver can decode the received codeword unambiguously

---

This work partially supported by Educational Mission of Saudi Arabia and the National Science Foundation under Contract No. NCR-9057872

into the codewords of the T or less active users. Every active user randomly selects a code from the set of N_C codes. The code rate $R(T)$ of T or fewer active users out of the N_c total users must be less than or equal to the channel capacity ($R \leq C$). The capacity of any T out of N_c real adder channel is not known, but it is upper bounded by the capacity of the T-user real adder channel. For binary channel inputs and equal rates for all users, the symmetric capacity per individual user is given by [6]:

$$C(T) \; = \; \frac{1}{T}\sum_{i=0}^{T} \frac{\binom{T}{i}}{2^T} \log_2 \frac{2^T}{\binom{T}{i}} \tag{1}$$

If the number N of active users in any given slot is
(i) $N \leq T$ and all users selected distinct codes, then the feedback message will be "success".
(ii) $N > T$ or $N \leq T$ and at least two users selected the same code, then the feedback message will be "collision". The probability that all N users select distinct codes from a (N_c, T) MUC code (such that unique decoding is possible at the central receiver) is

$$P_N(N_c, T) \; = \; \begin{cases} N_c^{-N} \frac{(N_c)!}{(N_c-N)!}, \; if \; N \leq T \\ 0, \; if \; N > T. \end{cases} \tag{2}$$

If collisions occur, they will be resolved using a CRA.

The intuitive underlying idea is that small collisions among T or less users (e.g., $T = 2$ or $T = 3$) are "resolved" by using MUC's and large collisions (which are assumed to occur very infrequently) are resolved by using CRA's.

II. An Example : Blocked-Access Tree Algorithm

The **Tree algorithm** is a splitting algorithm for resolving collisions between users transmitting in the same slot. When a collision occurs, all users which are involved in the collision have to flip a fair Q-sided coin to split into Q subsets. The first subset then transmits immediately in the next slot while the remaining $Q-1$ subsets have to wait until collisions in the first subset have been resolved (for details see [4]). If new packets must wait until all collisions pending at the time of their arrival to the system are completely resolved, then this protocol is called *blocked-access* algorithm.

In the following, we sketch the procedure for the throughput computation for the binary feedback (collision nocollision) blocked-access HRAS .

We use the same approach as in [4] by deriving

1. The probability generating function (PGF) $G_N(s) = E[s^{Y_N}]$ of the collision resolution interval (CRI) of length Y_N, given N packets initially collided. The CRI is defined as the time span from the beginning of a slot where an initial collision occurs up to and including the slot from which all active users recognize that all packets which were involved in the above initial collision have been successfully received. This PGF can then be used to obtain the moments of Y_N, in particular, $L_N = E[Y_N]$, which is the expected CRI length given N packets collided.

2. Using an exponential generating function (EGF) for L_N to get a description of the moments of the CRI length by functional equations.
3. Analysis of the asymptotic behavior of L_N as $N \rightarrow \infty$, which is essential for the determination of the maximum stable throughput λ^c_{crit}.

Theorem 1

The maximum stable throughput of the blocked-access HRAS is given by the following equation;

$$\lambda^c_{\text{crit}} = \left[\frac{lnQ}{Q - \sum\limits_{k=2}^{T} a_k \Gamma(k-1)} \right] \cdot R(T) \qquad (3)$$

where;

$$a_k = P_k(N_c, T) \sum_{j=1}^{Q} \sum_{i_j=0}^{k} \binom{k}{i_j} P_j^{i_j} (1 - P_j)^{k-i_j} \frac{L_{i_j}}{k!} \qquad (4)$$

L_{i_j} is given recersively by;

$$L_N = 1 + (1 - P_N(N_c, T)) \sum_{j=1}^{Q} \sum_{i_j=0}^{N} \binom{N}{i_j} P_j^{i_j} (1 - P_j)^{N-i_j} L_{i_j}; \qquad N \geq 2 \qquad (5)$$

$$P_j \triangleq \text{Prob. (the } j^{\text{th}} \text{ value of the "coin" flipped), with } \sum_{j=1}^{Q} P_j = 1 \qquad (6)$$

and $P_N(N_c, T)$ is given in (2). □

Proof: The proof of this theorem is rather lengthy and will be omitted (the complete analysis is given in [8]). The idea is to evaluate a_k recersivly for $k \leq T$, then subistitute in (3) to get λ^c_{crit}. ■

Using the capacity of the T-user real adder channel as an upper bound (1) and the codes given in [5] as a lower bound, Figure 2 shows the maximum stable throughput of HRAS's(7,3) λ^c_{crit} (upper and lower bounds) together with the maximum stable throughput of the uncoded RAS's λ_{crit} versus Q .

III. Upper Bound for the Capacity

The capacity of the HRAS is defined as the upper bound of the maximum throughput. Here we give an upper bound for the capacity of a HRAS as a function of the code parameters (N_c,T, $R(T)$). The superposition of all the users arrival processes is a poisson process denoted by X^* with finite rate λ (packets/slot). Successive instants of appearance of packets are denoted by $x_1^* \leq x_2^* \ldots \leq x_i^* \ldots$, $x_i^* \in \Re$. Coded packets can be transmitted by users only on time slots $(t, t+1)$, $t \epsilon \{0, 1, 2, \ldots.\} = I_0$. In any slot $(t, t+1)$, one of $T+2$ feedback messages θ_t may occur, which are:

1. $\theta_t = 0$, empty slot
2. $\theta_t = i$, $1 \leq i \leq T$ successful transmission of i packets
3. $\theta_t = T + 1$, collision occured

As described in section I, success occurs if the number N of active users $N \leq T$ and all users selected distinct codes, while collision occurs if the number N of active users $N > T$ or $N \leq T$ and at least two users selected the same code.
At the beginning of each slot, each user of the system is made aware of which of the $T + 2$ events has occured in the preceding slot. The users store this information, so that at the beginning of any slot $[t, t+1)$, each user knows the events $\theta(t) = (\theta_1, \ldots, \theta_t)$ that have taken place in all preceding slots. The transmission algorithm μ^* is specified by an infinite $T + 1$-ary tree [3]. The vertices of the tree are numbered by words z, $z \in Z$, where Z is the set of all sequences of length $|z|$, $|z| \in I_1$, made up of numbers $\{0, 1, \ldots, T + 1\}$. Every vertex with number z is associated with set $B[z], z \in Z$, $B[z] \subset \Re_+$. Any sequence of events $\theta(t) = (\theta_1, \ldots, \theta_t)$ corresponds uniquely to a path on the tree, hence it corresponds to a single value of the set $B[z] = B[\theta_1, \ldots, \theta_t]$. Any user will transmit a packet at time t if and only if the instant of appearance of its packet belongs to the set $B[\theta_1, \ldots, \theta_t]$.

Now, we are interested in the number of points x_i^* of stream X^* that appear in some set B. At time $t = 1$, algorithm $f^* \in \mu^*$ checks how many points of stream X^* belong to some set B_1. The result of this check is $\theta(B_1)$. If $\theta(B_1) \leq T$, we discard from X^* those points that belong to B_1, and subsequent checks involve only the remaining undiscarded points of X^*. Algorithm f^* is defined as an infinite sequence of checks of sets from $\Re_+$.
Define D_{t-1} as

$$D_{t-1} = \bigcup_{\substack{l=1 \\ \theta(B_l) \leq T}}^{t-1} B_l \tag{7}$$

D_{t-1} is called the set that is inspected up to time t, i.e., by time t, all points belonging to D_{t-1} are discarded from stream X^*.
For any $f^* \in \mu^*$ and $A \in \Re_+$, define the length of the interval $\tau(f^*, A)$ as:

$$\tau(f^*, A) = min\{t : [0, A] \supset D_{t-1}\} \tag{8}$$

By $\tau(f^*, A)$ all points belonging to $(0, A)$ are discarded from X^*. The rate of algorithm f^* is defined as:

$$R_{f^*} = \lim_{A \to \infty} \frac{A}{E[\tau(f^*, A)]} \tag{9}$$

The capacity is then defined as:

$$C^* = \sup_{f^* \in \mu^*} R_{f^*} \tag{10}$$

The above descreption for algorithm $f^* \in \mu^*$ is called a time-developing stream (TDS) [3] if it satisfies the following condition

$$B[\theta_1, \ldots, \theta_t] \subset [0, t) \tag{11}$$

The above condtion means that packets which have not yet appeared cannot be transmitted. A ready stream (RS) algorithm $f \in \mu$ is an algorithm that doesn't satisfy the above condition and has a combined poisson process X with rate $\lambda = 1$.
Defintions (7)-(10) hold for the RS algorithm f.

Theorem 2

Assume that some RS algorithm f has rate $R_f > 0$ and $E[\tau^2(f, A)] < \infty$,for any $A < \infty$, then there exists a TDS algorithm f^* with rate $R_f^* = R_f$. □

Proof: Omitted (see theorem 1 in [3]) ∎

Theorem 3

The capacity of the TDS algorithm is upper bounded by the capacity of the RS algorithm

$$C^* \leq C$$

□

Proof: Omitted (see theorem 2 in [3]) ∎

In what follows, we will find an upper bound for the capacity C of the RS algorithm. Note that, due to coding, the capacity depends on the code parameters $(N_c, T, R(T))$. Hence, the capacity will be denoted by $C(N_c, T)$ instead of C.
The main idea is to find an upper bound for the conditional probability of the event $\theta(B_t) = i$ and use it to bound the capacity.

Lemma 1

For any instant t , $t \in I_1$ and results of checks $\theta(t-1) = (\theta(B_1), ..., \theta(B_{t-1}))$ specified in algorithm f, $f \in \mu$, there exists a monotonically nondecreasing sequence y_j, $j = 0, 1, ..., T$, $0 \leq y_j \leq 1$, that for any $i = 0, 1, ..., T$,

$$Pr\{\theta(B_t) = i/\theta(t-1)\} \leq y_i p_i P_i(N_c, T)[\sum_{j=0}^{T-1} y_j p_j + y_T(1 - \sum_{j=0}^{T-1} p_j)]^{-1} \tag{12}$$

where

$$p_i = a^i e^{-a}/i!, \; i = 0, 1, ..., T-1;$$

$$a = mes(B_t \setminus D_{t-1}), \text{ is the Lebesuge measure of the set } B_t \setminus D_{t-1}$$

and $P_i(N_c, T)$ is given in (2) □

Proof: The proof of this lemma is analogous to the proof of lemma 2 in [3] except that

$$Pr\{\theta_t = i\} \;=\; \begin{cases} p_i P_i(N_c, T), & \text{i=0,1,...T} \\ 1 - \sum_{j=0}^{T} p_j P_j(N_c, T), & \text{i=T+1} \end{cases} \tag{13}$$

■

Theorem 4

We have the following upper bound for the capacity of a HRAS:

$$C(N_c, T) \;\leq\; \min_{q>0} \sup_{a>0} \max_{k} [(\sum_{i=k}^{T} \frac{P_i(N_c, T) a^i}{i!}(a + iq)) / \sum_{i=k}^{\infty} \frac{a^i}{i!}(1 + q)].R(T) \tag{14}$$

where the minimum is taken with respect to positive numbers q, the upper bound with respect to positive real numbers a, and the maximum with respect to $k \in \{0, 1, ..., T\}$,$P_i(N_c, T)$ is given by (2) and $R(T)$ is the code rate per user. □

Proof: conceptually, The proof of this theorem is similar to the proof of Thoerem 4 in [3], except for considering the code rate and the random code selection by active users (the proof is given in [8]). ■

A lower bound for the CRA part of the HRAS's is given in [7], using the part-and-try algorithm for retransmission.

Figure 3 shows the upper and lower bounds for the capacity of HRAS's using (7,T) codes.

IV. Conclusions

We mostly considered short codes (small N_c) because such codes are relatively easy to design and long codes have longer roundtrip delays. ¿From Figure 3, we can see that using (N_c, T) codes in combination with Random-Access Systems improves the system capacity. Capacity is maximum when $T = 3$ for all (N_c, T) codes. Thus, the search for good MUC's for HRAS's can be considerably narrowed, since large T offer no advantage at all.

References

[1] R.G. Gallager, "A perspective on Multi-Access Channels." IEEE Trans. Info. Theory, vol. IT-31, No. 2, pp. 124-142, Mar. 1985.

[2] V.A. Mikhailov and B.S. Tsybakov " Upper bound for the Capacity of a Random Multiple Access system", Probl. peredechi Inform. , vol. 17, No. 1, pp. 90-95, Jan.-Mar., 1981.

[3] B.S. Tsybakov, V. A. Mikhailov, and N. B. Likhanov, " Bounds for Transmission Rate in a Random-Multiple-Access System" Probl. peredechi Inform. , vol. 19, No. 1, pp. 61-81, Jan-Mar, 1983.

[4] P. Mathys and P. Flajolet, " Q-ary collision resolution algorithms in Random-access System with Free or Blocked Channel Access," IEEE Trans. Info. Theory, vol. IT–31, No. 2, pp. 217-243, Mar. 1985

[5] P. Mathys, "A Class of Codes for a T Active Users Out of N Multiple-Access Communication System," IEEE Trans. Info. Theory, vol. 36, No. 6, pp. 1206-1219, Nov. 1990

[6] S.C.Chang and E.J. Weldon, Jr., "Coding for T User Multiple Access Channels," IEEE Trans. Info. Theory, vol. IT-25, No. 6, pp. 684-691, Nov. 1979.

[7] R. M. AL-Rumaih and P. Mathys "Analysis of 0.487 Collision Resolution Algorithm with Multi-User Coding". presented at the 27^{t}h Annual Conf. on Info. Sciences and Systems, Baltimore, Maryland, Mar 24-26, 1993

[8] R. M. AL-Rumaih and P. Mathys, "Analysis of Random-Access Systems with Multi-User Coding", DSP-511 Techical Report , Department of Electrical and computer Engineering, University of Colorado, Boulder,CO 80309.

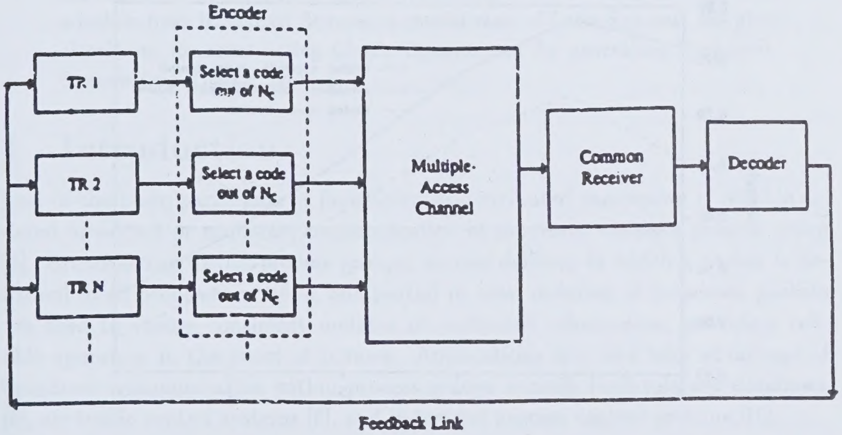

Encoder
TR 1
Select a code out of N_c
TR 2
Select a code out of N_c
TR N
Select a code out of N_c
Multiple-Access Channel
Common Receiver
Decoder
Feedback Link

Figure 1: System block diagram

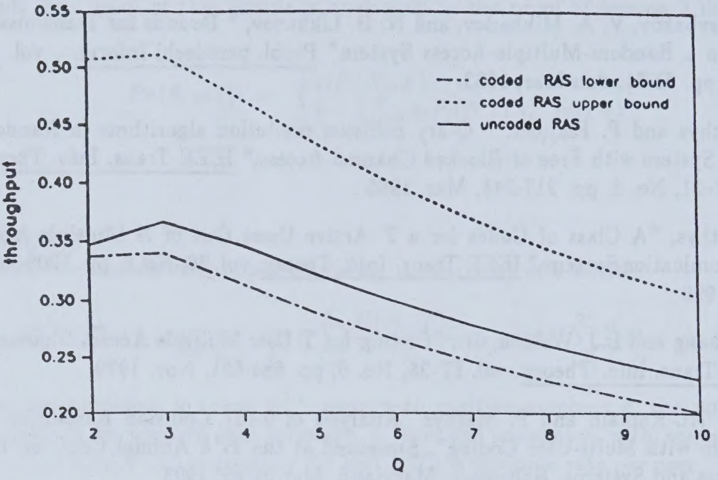


Figure 2: Throughput for coded (7,3) and uncoded blocked-access RAS

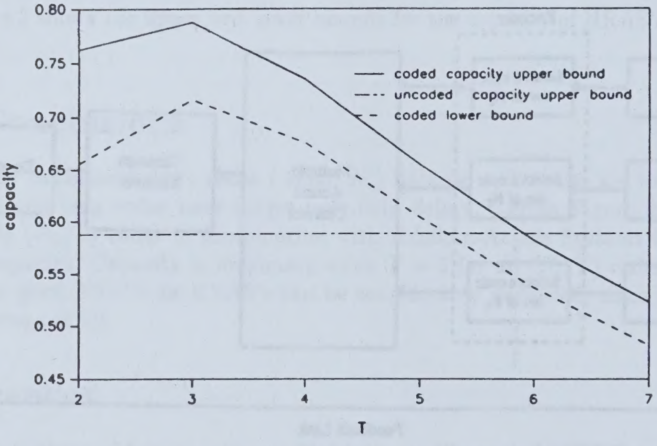


Figure 3: Upper and lower bounds for the capacity of HRAS's using $(7, T)$ codes

Scheduling Broadcast Communication on a High-Speed Network

M. Valerio, L. E. Moser and P. M. Melliar-Smith
Department of Electrical and Computer Engineering
University of California, Santa Barbara 93106

Abstract. We present two methods of implementing broadcast or multicast communication on a high-speed network that provides point-to-point packet transmission. Buffer overflow at the destinations is a major cause of packet loss on high-speed broadcast networks. Because of packet loss and retransmission, the rate of useful broadcasting may be much less than could be sustained by either the medium, the source, or the destinations. To avoid packet loss due to buffer overflow, we schedule the transmission of packets so that no two nodes transmit a packet to the same node at any time. The first method we present is a store-and-forward mechanism that requires processing capability of the nodes. The second method constructs a transmission schedule from Broadcast Squares, a special class of Latin Squares. We give algorithms for constructing Classic Squares and for generating Broadcast Squares from Classic Squares.

1 Introduction

One of the basic paradigms of fault-tolerant distributed computing is reliable ordered broadcast or multicast communication to processes within a process group [4]. Broadcasting within process groups, atomic delivery in which a packet is delivered to all processes or none, and partial or total ordering of broadcast packets are used to ensure consistent updates of replicated information, providing reliable operation in the event of failures. Applications that can take advantage of broadcast communication within process groups include fault-tolerant databases [5], air-traffic control systems [6], and industrial process control systems [10].

Extensive efforts have been undertaken to devise efficient protocols for ordered broadcasts within process groups that exploit the physical broadcast capability of local area networks [1, 2, 4, 12]. Typically, transmission over such networks is quite

---

This research was supported in part by NSF grant NCR-9016361 and by a grant from Apple Computer Corporation through the State of California Micro Program.

reliable; the primary cause of retransmission is buffer overflow at the destinations, rather than corruption of packets by the medium. It is evident that the maximum rate at which packets can be broadcast is the rate at which packets can be received and processed by the slowest node. Consequently, broadcast protocols must contain effective flow control mechanisms to ensure that the slowest node is not inundated with packets.

High-speed networks that operate in excess of 100 Mbit/s typically provide point-to-point packet transmission rather than broadcast communication [3, 9, 11]. With such communication media, a broadcast must be implemented as a set of point-to-point transmissions. High-speed networks can communicate packets many times faster than packets can be transmitted by a single node and many times faster than packets can be received and processed. If several sources transmit packets point-to-point to the same destination at the same time, that destination may be unable to receive and process those packets fast enough to avoid packet loss due to buffer overflow.

The objective of this paper is to show how broadcast communication can be implemented using point-to-point packet transmissions that are scheduled to ensure that no node must receive and process packets faster than it is able. All of the other services required of a reliable ordered broadcast protocol can then readily be provided, including retransmission of corrupted and lost packets, recovery from node failure, remerging of the network after partitioning, and other types of faults as in [2].

This paper is organized as follows. In Section 2 we describe the network model, and in Section 3 we provide two methods of implementing broadcast communication without message loss due to buffer overflow. The first method is a store-and-forward mechanism that requires processing capability of the nodes. The second method has no such requirement but places the burden of broadcasting on the node originating the broadcast. It constructs a transmission schedule from Broadcast Squares that are generated from Classic Squares. Sections 4 and 5 contain algorithms for generating Broadcast Squares from Classic Squares and for generating Classic Squares. Section 6 contains concluding remarks.

2 The Network Model

We consider n nodes, numbered $0, 1, \ldots, n-1$, connected by a communication network that provides point-to-point packet transmission between every pair of nodes. The physical topology of the network may not actually be completely connected in the sense that there is a physical link between every pair of nodes. The network is assumed to have high performance so that many packets can be transmitted point-to-point concurrently without interference. This assumption is not unrealistic if we compare the bandwidths of networks like SCI, ATM and

FDDI (ranging in excess of 100 Mbits/s) with current processing rates (around 5-10 Mbits/s).

We assume that a node forwards any packet not addressed to it very quickly, which is the case for available high-performance networks. Since a node does not spend processing time on a packet not addressed to it, we do not need to be concerned about buffer overflow at intermediate nodes between the sender and the receiver. Based on these assumptions (which correspond to practical high-performance networks), we can look at the network logically as a complete graph, even though there might not be a physical path between each pair of nodes.

A node is assumed to be able to transmit only a single packet at a time and to be able to receive and insert into its input buffer a single packet at a time. Reception may be concurrent with transmission. The nodes are also assumed to be equipped with clocks that are approximately synchronized.

The scheduling mechanism is described below as though all nodes broadcast packets at the same rate and every node has a packet to transmit at every opportunity. The mechanism is, however, equally applicable to unequal rates of broadcasting and to nodes that do not always have a packet to transmit.

3 Implementing Broadcast Communication

The objective of our scheduling strategy is to allow nodes to broadcast packets at the maximum rate such that no node receives two packets simultaneously and no node's input buffer space overflows because the node must receive packets faster than it can process them.

3.1 First Method

The nodes in the system are organized into a logical ring, which may bear no relation to the physical topology of the network [2, 13]. This ring is operated as a token ring with a special bit in each data packet, called the token bit, that marks the packet as bearer of the token. A node receiving a packet marked this way sets the token bit of the data packet it sends in the next time slot. If it has no packet to send, then it sends a null packet with the token bit set.

This method requires that nodes wishing to broadcast do so in the same time slot. The token acts as a synchronizer, allowing the nodes to agree upon a time slot in which to begin a broadcast. We have chosen the time slot after the token reaches node 0 as this special time slot. Thus, the time slot at which a node begins its broadcast depends upon its identifier and on the last time it saw the token. For example, node 0 begins its broadcast immediately after receiving the token, node $n-1$ begins its broadcast one time slot after receiving the token and, in general, node i begins its broadcast $(n-i) \bmod n$ time slots after receiving the token.

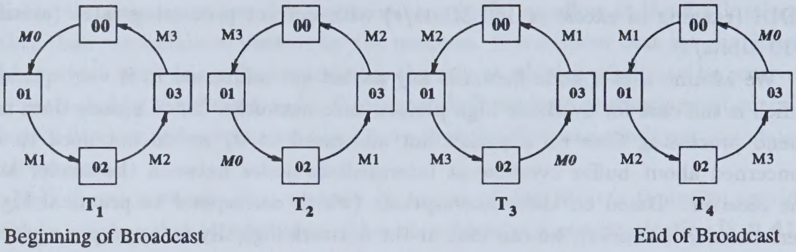


Figure 1: One Broadcast Cycle for Four Nodes.

On receiving a message marked for broadcast, a node copies it, and forwards it in the next time slot. The entire cycle ends after n time slots, when the originating node receives its own message back, notifying it that the broadcast was completed successfully. Figure 1 illustrates this in the case of a network with four nodes, the message in italic has its token bit set.

This method is quite straightforward and elegant, but it has the drawback that it requires all nodes in the network to have some processing capability. Not only must the nodes be able to process the token bit, but they must also be able to determine whether a packet is marked for broadcast. If the packet is so marked, a node must store the packet and then forward it to the next node on the ring. The rate at which it does this processing must be comparable to the rate at which the network transmits messages. Our second method, described below, puts the burden of broadcasting on the node performing it and, thus, does not require the processing capability of the receiver.

3.2 Second Method

As in the first method, the nodes in the system are organized into a logical ring, which may bear no relation to the physical topology of the network [2, 13]. This ring is operated as a token ring with a special packet, the token, circulating around the ring. As each node receives the token, it performs the following sequence of actions:

- Process the token and transmit it to the next node on the ring.

- Transmit one copy of the packet to each of the $n-1$ other nodes, according to the schedule specified below. The point-to-point transmission of these $n-1$ copies of a packet represents the broadcasting of one packet.

Time	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Node 0	T_1	3	2	1	T_1	3	2	1	T_1	3	2	1	T_1	3	2	1
Node 1	-	T_2	0	2	3	T_2	0	2	3	T_2	0	2	3	T_2	0	2
Node 2	-	-	T_3	3	0	1	T_3	3	0	1	T_3	3	0	1	T_3	3
Node 3	-	-	-	T_0	2	0	1	T_0	2	0	1	T_0	2	0	1	T_0

Figure 2: A Packet Transmission Schedule That Avoids Buffer Overflow.

The token can be exploited to achieve other properties needed for reliable ordered broadcast communication, including acknowledgment of packets, determination that packets have been received by all destinations, retransmission of lost or corrupted packets, ordering of packets, flow control, detection of and recovery from node failure and network partitioning [2].

The most important aspect of the broadcast mechanism presented here is the scheduling of point-to-point packet transmissions. Time is divided into slots using approximately synchronized clocks; within a time slot a node can transmit one packet. By carefully selecting the order in which packets are transmitted, we can guarantee that no two nodes will transmit a packet to the same destination at any time.

For example, if there are four nodes on the ring, we can choose to transmit packets according to the packet transmission schedule shown in Figure 2. This schedule indicates, for example, that in the fifth clock cycle node 0 transmits the token to node 1, node 1 transmits a packet to node 3, node 2 transmits a packet to node 0, and node 3 transmits a packet to node 2. As is evident, no two nodes transmit to the same node at any time and, thus, no receiver will have buffer overflow problems due to multiple concurrent receptions.

Multicasting can be achieved by transmitting to the nodes in the desired set. If one or more nodes do not have a message to broadcast, they may skip one or more cycles. If the nodes have different communication rates, all of them must be slowed down to match the rate of the slowest node. This is not a drawback, since the maximum rate at which packets can be broadcast without buffer overflow is limited by the slowest node's rate of processing packets.

It is important to note that each node must wait for the token before broadcasting a new packet and that each node transmits each packet marked for broadcast to each of the other nodes. The pattern of transmissions is, thus, periodic with period n. Figure 3 represents the repeated pattern of the packet transmission schedule of Figure 2 more compactly. The token entries T_i along the diagonal of the square in Figure 3 represent the token transmitted by node $i-1$ to node i. In our analysis, we do not need to distinguish between the token and ordinary packets. The token, like an ordinary packet, contains information that requires processing. Consequently, we can replace each entry T_i by the element i. After

T_1	3	2	1
3	T_2	0	2
0	1	T_3	3
2	0	1	T_0

Figure 3: A More Compact Representation of the Schedule of Figure 2.

this replacement, we obtain the square shown in Figure 4 below. If we number the rows and columns $0, 1, \ldots, n-1$ and denote the entries by a_{ij}, $0 \leq i, j < n$, then this square has the following interesting properties.

BS1: In row i, element i is missing, element $i+1$ is duplicated, and each of the other elements appears exactly once.

BS2: The elements in each column are distinct.

BS3: The elements on the main diagonal are $1, 2, \ldots, n-1, 0$,
i.e. $a_{ii} \equiv (i+1) \bmod n$ for $0 \leq i < n-1$.

Squares with these properties are referred to as *Broadcast Squares* of order n.

1	3	2	1
3	2	0	2
0	1	3	3
2	0	1	0

Figure 4: A Broadcast Square of Order 4.

4 Generating Broadcast Squares

In trying to devise an algorithm to construct Broadcast Squares, we found property BS1 difficult to ensure. Thus, instead of trying to construct Broadcast Squares directly, we found it easier to define another class of squares, called *Classic Squares*, and then to transform these squares into Broadcast Squares. Classic Squares have the following properties:

CS1: The elements in each row are distinct.

CS2: The elements in each column are distinct.

CS3: The elements on the main diagonal are $0, 1, 2, \ldots, n-1$,
i.e. $a_{ii} \equiv i \bmod n$ for $0 \leq i < n$.

CS4: The elements on the superdiagonal are $n-1, 0, 1, \ldots, n-2$,
i.e. $a_{ij} \equiv (i-1) \bmod n$ for $0 \leq i < n$ and $j \equiv (i+1) \bmod n$.

Properties CS1 and CS2 are the defining properties of Latin Squares [7]. An example of a Classic Square of order 6 is shown in Figure 5.

0	5	3	4	1	2
2	1	0	5	3	4
3	4	2	1	5	0
5	0	4	3	2	1
1	2	5	0	4	3
4	3	1	2	0	5

Figure 5: A Classic Square of Order 6.

Given an algorithm for generating Classic Squares, such as that described below in Section 5, Broadcast Squares of order n can be generated as follows:

- Choose a Classic Square of order $n-1$ as the upper left subsquare of the square of order n being constructed.
- Copy the elements from the superdiagonal of this Classic Square into their corresponding columns in the $(n-1)$st row of the square being constructed, *i.e.* $a_{n-1,j}$ becomes a_{ij}, where $0 \leq i < n-1$ and $j \equiv (i+1) \bmod (n-1)$.
- Replace all of the superdiagonal elements of the Classic Square by $n-1$, *i.e.* a_{ij} becomes $n-1$, where $0 \leq i < n-1$ and $j \equiv (i+1) \bmod (n-1)$.
- Interchange $a_{0,1}$ and $a_{n-1,1}$.
- Assign $0, 1, \ldots, n-1$ to the last column.
- Add 1 to each of the elements of the square and reduce modulo n.

Figure 6 shows the construction of a Broadcast Square of order 4. The upper left subsquare of the first square is a Classic Square of order 3.

0	2	1	
2	1	0	
1	0	2	

→

0	2	1	
2	1	0	
1	0	2	
1	2	0	

→

0	3	1	
2	1	3	
3	0	2	
1	2	0	

→

0	2	1	
2	1	3	
3	0	2	
1	3	0	

→

0	2	1	0
2	1	3	1
3	0	2	2
1	3	0	3

→

1	3	2	1
3	2	0	2
0	1	3	3
2	0	1	0

Figure 6: The Construction of a Broadcast Square of Order 4.

5 Generating Classic Squares

The algorithm for generating a Classic Square begins by finding all entries $a_{ij} \equiv (2i-j) \bmod n$, where $0 \leq i, j < n$. If n is odd, this simple algorithm yields a Classic Square.

If n is even, the square obtained from this formula does not satisfy property CS2; however, we can use this formula to construct a square, called a *Pre-Classic Square*, as follows:

- The lower half (the last $n/2$ rows) duplicates the upper half (the first $n/2$ rows), as depicted below.
- The entries in the upper half are given by $a_{ij} \equiv (2i - j) \bmod n$, where $0 \leq i < n/2$ and $0 \leq j < n$.

Elements on the diagonal and on the superdiagonal of a Pre-Classic Square are called *taboo* elements.

Figure 7 depicts a Pre-Classic Square. This square satisfies properties CS1, CS3 and CS4. It also satisfies the property that all even columns contain only even elements and all odd columns contain only odd elements.

$a_{0,0}$	$a_{0,1}$	$\cdots$	$a_{0,n-1}$
$a_{1,0}$	$a_{1,1}$	$\cdots$	$a_{1,n-1}$
$\vdots$			
$a_{\frac{n}{2}-1,0}$	$a_{\frac{n}{2}-1,1}$	$\cdots$	$a_{\frac{n}{2}-1,n-1}$
$a_{0,0}$	$a_{0,1}$	$\cdots$	$a_{0,n-1}$
$a_{1,0}$	$a_{1,1}$	$\cdots$	$a_{1,n-1}$
$\vdots$			
$a_{\frac{n}{2}-1,0}$	$a_{\frac{n}{2}-1,1}$	$\cdots$	$a_{\frac{n}{2}-1,n-1}$

Figure 7: A Pre-Classic Square.

A Pre-Classic Square does not satisfy property CS2 because each column contains duplicate elements, namely $a_{ij} = a_{i+\frac{n}{2},j}$. To satisfy property CS2, we interchange $n/2$ pairs of elements in adjacent columns to eliminate duplicates. That is, we convert a Pre-Classic Square into a Classic Square, as follows:

- Interchange either a_{ij} and $a_{i,j+1}$ or $a_{i+\frac{n}{2},j}$ and $a_{i+\frac{n}{2},j+1}$, where $0 \leq i < n/2$ and j is even, provided that the elements being interchanged are not taboo elements.

Interchanging these elements produces a square that satisfies property CS2. Figure 8 gives an example of the elements that must be interchanged to transform a Pre-Classic Square into a Classic Square for $n = 6$. The taboo elements are marked with the superscript t.

This algorithm generates Classic Squares for any odd number n and for even numbers n greater than 5. It cannot be used to generate Classic Squares of order

0^t	5^t	4 ←	→ 3	2 ←	→ 1
2	1^t	0^t	5	4 ←	→ 3
4 ←	→ 3	2^t	1^t	0 ←	→ 5
0 ←	→ 5	4	3^t	2^t	1
2 ←	→ 1	0 ←	→ 5	4^t	3^t
4^t	3	2 ←	→ 1	0	5^t

Figure 8: Transforming a Pre-Classic Square into a Classic Square for $n = 6$.

$n = 2$ or $n = 4$, because there are no such squares. Even though there are no Classic Squares of order 4, there do exist Broadcast Squares of order 5; however, there do not exist Broadcast Squares of order 3.

6 Conclusion

We have presented two methods for scheduling broadcast communication on a high-speed network with point-to-point packet transmission. Buffer overflow, the main cause of packet loss in high-speed networks, is avoided by restricting the transmission of packets so that no two nodes transmit packets to the same destination at any time. Both methods utilize a logical token passing ring. The first method requires all nodes in the network to have some processing capability and to be very fast at processing. The second method, which avoids these requirements, uses a transmission schedule based on Broadcast Squares that are derived from Classic Squares. Algorithms for generating Classic Squares and for transforming them into Broadcast Squares have been given.

References

[1] Amir, Y., Dolev, D., Kramer S., and Malki, D., "Transis: A communication sub-system for high availability," *Proceedings of the 22nd Annual International Symposium on Fault-Tolerant Computing*, Boston, MA, July 1992, pp. 76–84.

[2] Amir, Y., Moser, L. E., Melliar-Smith, P. M., Agarwal, D. A., and Ciarfella, P., "Fast message ordering and membership using a logical token passing ring," *Proceedings of the 13th International Conference on Distributed Computing Systems*, Pittsburgh, PA, May 1993, pp. 551–560.

[3] Minznern, S. E., "Broadband ISDN and Asynchronous Transfer Mode (ATM)," *IEEE Communications Magazine* 27 (1989), pp. 17–24.

[4] Birman, K. P., and Joseph, T. A., "Reliable communication in the presence of failures," *ACM Transactions on Computer Systems* 5, 1 (February 1987), pp. 47–76.

[5] Chang, J. M., "Simplifying distributed database systems design by using a broadcast network," *Proceedings of ACM SIGMOD '84* 14, 2, Boston, MA, June 1984, pp. 223–233.

[6] Cristian, F., Dancey, R. D., and Dehn, J., "Fault-tolerance in the Advanced Automation System," *Proceedings of the 20th Annual International Symposium on Fault-Tolerant Computing*, Newcastle-upon-Tyne, UK, June 1990, pp. 6–17.

[7] Dénes, J. and Keedwell, A. D., *Latin Squares and Their Applications.* Academic Press, New York, 1974.

[8] Dénes, J. and Keedwell, A. D., *Latin Squares: New Developments in the Theory and Applications.* Annals of Discrete Mathematics 46, Elsevier Science Publishers, Amsterdam, 1991.

[9] Gustavson, D. B., "The scalable coherent interface and related standards projects," *IEEE Micro* 12, 1 (February 1992), pp. 10–22.

[10] Kopetz, H., Damm, A., Koza, C., Mulazzani, M., Schwabl, W., Senft, C., and Zainlinger, R., "Distributed fault-tolerant real-time systems: The Mars approach," *IEEE Micro* (February 1989), pp. 25–40.

[11] Sweazey, P., QuickRing protocol specification. Version 0.921, Advanced Technology Group, Apple Computer Inc., September 1992.

[12] Melliar-Smith, P. M., Moser, L. E. and Agrawala, V., "Broadcast protocols for distributed systems," *IEEE Transactions on Parallel and Distributed Systems* 1, 1 (January 1990), pp. 17–25.

[13] Rajogapalan, B. and McKinley, P. K., "A token-based protocol for reliable, ordered multicast communication," *Proceedings of the 8th IEEE Symposium on Reliable Distributed Systems*, Seattle, WA, October 1989, pp. 84–93.

Delay Analysis of a Circuit-Switched Interconnection Network Implementing a Gated Hold Strategy

Paul Dietrich and Ramesh R. Rao *

University of California, San Diego
La Jolla, CA 92093

Abstract

In this paper, we analyze a circuit-switched, blocking, multistage interconnection network (MIN). We do not assume that blocked devices drop their requests nor that they re-randomize their destinations. A hold protocol is proposed that retains partial path information to speed routing of blocked messages. By formulating a dominating system, we are able to derive a recursive expression for the mean time to process a batch of requests. This yields an upper bound on message transfer time and a lower bound on message throughput of the actual system. This gated hold strategy is compared to a non-blocking network with a similar gated strategy, and results indicate that a blocking network implementing such a hold strategy outperforms a non-blocking network of comparable complexity for small message transfer times.

1 Introduction

In the design of interconnection networks, crossbar networks offer full connectivity between inputs and outputs but they are costly to build and impractical for a large number of inputs. Research, has shown [2], [1], [7], that blocking networks provide a better cost/performance ratio than crossbar networks. In blocking networks, a fraction of the messages are blocked due to contention within the switching fabric and must be resubmitted to guarantee reliable communications. Because requests may have to be submitted several times, a request experiences a random delay depending on the current state of the MIN. This exacerbates performance analysis.

In his pioneering analysis, Patel [2] ignores the resubmission of blocked requests and derives an expression for the bandwidth of a Banyan network. Kruskal and Snirs' analysis, [1], also assumes that dropped packets are not resubmitted. Analyses that ignores

---

*Supported by NSF under grant NCR-8904029

blocked requests cannot answer questions concerning message set-up delay. Bhattacharya , Rao and Lin [6] derived an upper bound on message delay for a synchronous circuit-switched delta network that accommodates the resubmission of blocked requests. To preserve uniformity and independence of the message input distribution, blocked requests were resubmitted after a random number of cycles. One deficiency of their model is the assumption that resubmitted requests re-randomize their destinations.

In this paper, we consider a synchronous, circuit-switched, self-routing Banyan network of 2 × 2 crossbar switches in which blocked devices resubmit their requests without re-randomizing their destinations. A critical aspect of this work is the formulation of a simple gated hold protocol, in which requests that are blocked during a contention period hold their partial paths and systematically advance themselves to their destination after non-blocked users ahead of them have finished. Because this protocol accommodates dropped requests, meaningful expressions for delay and throughput can be defined and are derived.

Another significant aspect of this work is the formulation of a worse case, dominating system which is easier to analyze. We derive a recursive equation for the expected time to service a set of requests at the input to the dominating system. The analysis produces an upper bound on message delay and a lower bound on message throughput for messages of equal length d for the actual system. In addition, the complexity of the recursive solution does not grow rapidly as network size increases, and delay/throughput results can be easily computed for network with N larger than 2^{16}.

Section 2 presents the interconnection network model along with the underlying assumptions. In section 3, the key points of the analysis are presented. In section 4 a simple analysis of a non-blocking network is derived for comparison purposes. Several bounds are investigated for the exact analysis of this non-blocking network when the network size is large. Section 5 presents the numerical results from the analysis. The derived delay upper bound is compared to a simulation of the actual system as well as to the delay of a non-blocking network. The final section summarizes the results and outlines future work.

2 Proposed Protocol

2.1 Protocol Description

We consider a synchronous circuit-switched network with a holding strategy. Time is divided into periods. At the beginning of each period, the input devices submit their request for connection to the output devices. As these request propagate through the MIN, some are blocked and some survive. The requests that are blocked hold their partial paths. When the requests that did not get blocked finish their jobs, the blocked jobs continue, starting with those in the stage of the MIN closest to the outputs of the network. This process continues until all requests have been served in which case a new period begins and the processors submit new requests that may have arrived. Unlike the synchronous circuit-switched protocol with request resubmission or dropping, these service periods are not of equal length, and depend upon the number of collisions that occur during contention for the communications paths. Such a service period will be called a network cycle.

This synchronous holding protocol retains the information gained in collisions about destination locations of messages. It would seem that a protocol that uses such information might yield better performance than one which discards information concerning collisions.

2.2 Assumptions

We chose a self-routing square Banyan network with 2 × 2 crossbar switches for the interconnection network. The self-routing property of these networks make them desirable for high-speed processor applications. In addition, we exploit the properties of these networks described in Kruskal and Snir [1].

If requests are submitted independently and uniformly at the beginning of each cycle with marginal probability p and uniformly distributed among all outputs independent of other requests then:

1) A request arriving at any switch in the MIN will require connection to the top output and bottom output with equal probability.

2) For any switch, the request arrival patterns are uniform and independent across switch inputs.

In addition we make the assumption that the cycle time required to service a network with no requests at its input is 0. Thus no time is spent in a cycle if the cycle has no initial requests.

3 Analysis

3.1 Construction of the Dominating System

Consider a self-routing square Banyan network constructed of 2 × 2 crossbar switches with N inputs. Such a networks requires k stages of switches where $N = 2^k$. At the beginning of each network cycle, the input devices submit requests independently with probability p. The destination of these requests are assumed to be uniformly distributed among all network outputs. It then follows [2],[1] that requests arriving at the inputs to a switch are equally likely to require connection to each output port. Also, requests arriving at any switch are independent of each other. It must be noted, however, that the requests departing from any stage are no longer independent, making the arrival process to the following stages dependent across all switches.

The exact analysis of such a system can be accomplished using a Markov chain description of the entire MIN. However, the number of states grows exponentially with the number of inputs making the analysis intractable for large networks. Instead, we compute a bound on performance of such a system by defining a worse case, dominating system. The dominating system is easier to analyze and we are able to derive a recursive equation that expresses the expected cycle time of a set of requests at the input to j remaining stages of an N-input network as some function of the expected cycle time of a set of requests at the input the $j-1$ remaining stage of the same N-input network.

Consider a dominating system that adds 'dummy' requests at the outputs of the switches to make the output request distributions uniform and independent. Of course, in the true system operation, no 'dummy' requests will be added to the requests. The throughput/delay of the actual system will, for every sample path, be higher/lower than predicted in the dominating system. One straightforward way to engineer a uniform and *independent* distribution at the output of stage one is to insert a dummy request on the empty outgoing link every time there is a collision within the switch. With such a switch, stage two would have the same input distribution as stage one. However, such an addition complicates analysis. In this analysis, we consider an addition strategy in which the switch, upon seeing a single request at its outputs, produces another request at its other output with a certain probability irrespective of whether or not blocking occurred within the switch. This addition strategy yields the same results as the previous strategy but, because it is not dependent upon whether contention occurred, makes the analysis more straightforward.

We also need to tailor the residual request so that they become uniform and *independent* across all inputs as well. To do this we let the residual propagate through one stage, since they will do so in a contention free manner. Upon observing a [0,1] or [1,0] at the output, the switch produces an extra dummy request along the empty path with probability α. We found that if

$$\alpha = \frac{1-\sqrt{1-\frac{p^2}{2}}}{\frac{p^2}{4}} - 1 \tag{1}$$

then the distribution at the output of that stage will be uniform and *independent.* Here p is assumed to be non-zero in which case α is a number between 0 and 1. In the dominating system, the residual requests will pass the first stage without contention and arrive at the input to the $k-1$ stage network with uniform and *independent* distribution with marginal probability given by:

$$T(p) \triangleq 1 - \sqrt{1-\frac{p^2}{2}} \tag{2}$$

3.2 Analysis of the Dominating System

Define a random variable $f^N(p,j)$ to be the number of switching cycles required to set up circuits for a set of request with independent and uniform distributions with marginal input probability p (i.e. the total time required to set up connections for all input devices requiring such connection) at the input to the remaining j stages of a square (N input) Banyan network. A switching cycle is defined as the time taken for a request to propagate through one stage of the switching network. Throughout this analysis, a switching cycle requires 1 unit of time. A rigorous derivation, details of which are omitted, revealed that

$$E[f^N(p,k)] = E[f^N(p,k-1)] + E[f^N(T(p),k-1)] + E[f^N(p,1)]. \tag{3}$$

With careful thought one could have written this relationship down directly. The switches at the input effectively sort the incoming requests into two groups so that no two request in a given group are destined to the same output port of the same switch. It requires $E[f^N(p,1)]$ switching cycles to pass these two groups to the second stage. In the dominating system, the two input distributions at the second stage of switches are uniform and independent. These two groups must pass through the remaining $k-1$ stages which requires a total of $E[f^N(p,k-1)] + E[f^N(T(p),k-1)]$ switching cycles.

Adding the message transmission or circuit hold time time d is not difficult, but leads to a slightly more complicated recursive equation. Consider incorporating the time d into the last stage of the request propagation. In other words, when the request travels through the last stage an additional delay of d is added due to the message delivery time. Define $g^N(p,j)$ to be the number of slots required to completely serve a set of request with independent and uniform distribution with marginal input probability p (i.e. set up links and transfer messages) at the input to the remaining j stages of a square (N input) Banyan network. One can derive a similar recursive equation for $g^N(p,k)$, and obtain

$$E[g^N(p,k)] = E[g^N(p,k-1)] + E[g^N(T(p),k-1)] + 2 - (1-p)^N - (1-\frac{p^2}{2})^{N/2}. \tag{4}$$

The initial step can be shown to be:

$$E[g^N(p,1)] = (d+1) * \left(2 - (1-p)^N - (1-\frac{p^2}{2})^{N/2}\right) \tag{5}$$

This is the recursive equation for the expected setup and delivery time, which can be solved for $E[g^N(p,k)]$. In the results section, this model will be compared to a simulation of the actual system (i.e. without adding the 'dummy' requests) and to a non-blocking switch with similar complexity.

4 Comparison with Non-Blocking Networks

To compare the performance of such a hold scheme for a blocking network, we examine the performance of a hold protocol for a non-blocking network. Since there is no internal blocking in these networks, the protocol reduces to a simple gated input strategy. At the beginning of each cycle, requests are generated with a certain probability p and passed into the network. If two or more messages are destined for the same output, only one is allowed to pass. The blocked requests resubmit in the next sweep without rerandomizing their destinations. No new requests enter the network until all current requests have completed service.

Consider and N-input, N-output non-blocking network. Define the vector X to be a vector of indicator random variables, each element of X taking value 1 with probability p independent of other elements. X represents the set of active inputs. Given a known number of active inputs U, if all messages were able to travel to the outputs simultaneously without collision, then the output vector Y ,with element Y_i defined as the number of active inputs with a message destined for output i, would have multinomial distribution with probability $p_i = 1/N$:

$$P(Y_1 = y_1, Y_2 = y_2, \ldots, Y_N = y_N) = \frac{U!}{y_1!y_2!\ldots y_N!}\left(\frac{1}{N}\right)^U \tag{6}$$

where

$$\sum_{i=1}^{N} y_i = U \tag{7}$$

Assuming, as in the Banyan analysis, that the messages do not rerandomize when blocked, the number of sweeps required per cycle to transmit the input messages in X given U is:

$$Z = \max(Y_1, Y_2, \ldots, Y_N) \tag{8}$$

In theory it is easy to find $F_{Z|U}(z)$. It is given by:

$$F_{Z|U}(z) = F_{Y_1,Y_2,\ldots,Y_N}(z, z, \ldots, z) \tag{9}$$

Unfortunately, it is computationally difficult to evaluate the above as the size of the network grows. Two techniques were considered for bounding the performance of the non-blocking switch with gated strategy assuming no rerandomizing of blocked request destinations. The first bound [11], although simple to compute, is too loose for valid comparison. The second [9], although much tighter, is computationally intensive and has little advantage over computing the actual quantity. The final bound [10], requires a moderate amount of computation, but yields unsatisfactory bounds. Therefore, for moderately large networks, a simulation of the non-blocking switch was used for comparison.

5 Results

Via computer , the recursive equation for the upper bound to the cycle time of a blocking network using a gated hold protocol can be computed for very large networks ($N > 2^{16}$) in short periods of time. Unfortunately, a simulation of such a large network becomes impractical due to limited resources. However, a simulation for ($N < 64$) shows that this bound is fairly tight for all values of p. In the simulation 'dummy' requests are not added. For ($p < .5$) the bound is excellent (See Fig. 1). Figure 2 shows the bound parameterized by d. For a fixed p, the bound increases almost linearly in d. Even for a relatively small choice of d, the performance of such a system is degraded. However, the performance of the non-blocking switch is also linearly increasing in d.

In Figs. 3, message throughput is shown versus p for various values of N. Message throughput is defined to be the average number of messages that complete service during each switching time. It should be noted that the bound is not monotonically increasing in p indicating that there may be a 'best choice' value of p to maximize the network throughput. A simulation for $N = 128$ failed to confirm this behavior in the actual system. This simulation revealed that maximum value of throughput for a value of $p < 1$ predicted by the upper bound is only present in the dominating system.

The set of graphs Fig. 4, and 5, compare the upper bound on blocking network performance to the simulated performance of a non-blocking switch using a gated input strategy. For comparison, the non-blocking network is chosen to be a Batcher-Banyan type. The Banyan switch requires k stages for and N input network while the Batcher-Banyan switch requires $k(k+1)/2+k$ stages for its sorting and routing operation. For $d=5$, the blocking network offers superior performance for almost all values of p. As d grows the bound for the blocking network exceeds the performance of the non-blocking network. However, it is inconclusive whether the actual system performance of the non-blocking network is worse.

6 Conclusions

In this paper a bound on the performance of a MIN implementing a gated hold strategy is derived. The bound is relatively tight for small values of p and is excellent for small N. However, as N grows, the bound becomes looser. Although this bound is loose for large N, it can be computed with little computational effort. Thus it provides at least some indication of system performance for large N when simulation becomes too costly.

This work is being extended to provide bounds on the performance of a circuit switched MIN with non-uniform input traffic. In addition, traffic with non-uniform output destinations is also being considered. Future work includes applying these techniques to the analysis of buffered packet switched networks as well.

The methodology developed in this paper appears to be rather promising in that it provides a relatively rich source of alternatives to approximating the stochastic evolution of these networks.

References

[1] Clyde P. Kruskal and Marc Snir, “The Performance of Multistage Interconnection Networks for Multiprocessors,” *IEEE Trans. on Comput.* , Vol C-32, N0 12, Dec. 1983, pp. 1091-1098.

[2] Janak H. Patel, “Performance of Processor-Memory Interconnections for Multiprocessors,” *IEEE Trans. on Comput.* Vol C-30, No. 10, Oct. 1981, pp. 771-780.

[3] Peter G. Harrison and Naresh M. Patel, “The Representation of Multistage Interconnection Networks in Queueing Models of Parallel Systems,” *Journal of the Association for Computing Machinery*, Vol 37, No. 4, Oct. 1990, pp. 863-898.

[4] Chuan-lin Wu and Manjai Lee, “Performance Analysis of Multistage Interconnection Network Configurations and Operations,” *IEEE Trans. on Comput.*, Vol-41, No. 1, Jan. 1992, pp. 18-26.

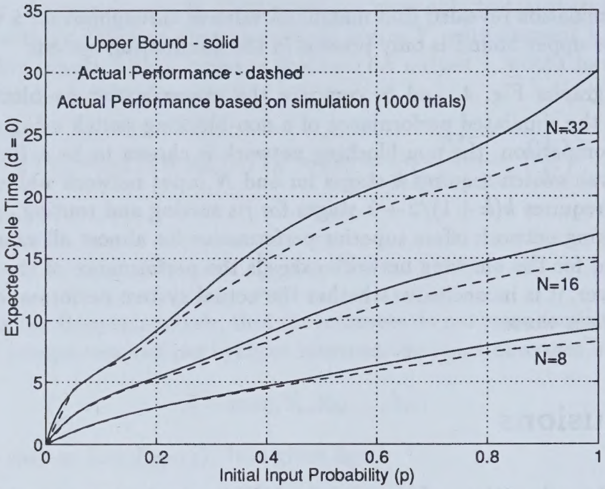


Figure 1: A comparison of actual network performance versus the computed upper bound for a blocking Banyan network using a gated hold strategy.

[5] Shuo-Hsien Hsiao and C. Y. Roger Chen, "Performance Evaluation of Circuit Switched Multistage Interconnection Networks Using A Hold Strategy," *IEEE Transactions on Parallel and Distributed Systems*, Vol. 3, No. 5, Sept. 1992, pp. 632 - 640.

[6] Amiya Bhattacharya, Ramesh R. Rao, and Ting-Ting Lin, "Delay Analysis of Synchronous Circuit Switched Delta Networks," *In Proceedings, 1992 International Parallel Processing Symposium*, Newport Beach, Ca.

[7] Clyde P. Kruskal and Marc Snir, "Cost Performance Tradeoffs for Interconnection Networks," *Discrete Applied Mathematics*, 37/38 (1992) pp. 359-385.

[8] Yuju Oie, Tatsuya Suda, etc., "Survey of Switching Techniques in High-Speed Networks and Their Performance," *International Journal of Satellite Communications*, Vol. 9, 1991, pp. 285-303.

[9] Barry C. Arnold and Richard A. Groeneveld, "Bounds On Expectations Of Linear Systematic Statistics Based On Dependent Samples," *The Annals of Statistics*, Vol 7. No. 1, 1979, pp. 220-223.

[10] Barry C. Arnold, "p-Norm Bounds on The Expectation of The Maximum of A Possibly Dependent Sample," *Journal of Multivariate Analysis*, Vol. 17, 1985, pp. 316 - 332.

[11] Donald R. Hoover, "Bounds On Expectations Of Order Statistics For Dependent Samples," *Statistics and Probability Letters* Vol. 8, 1989, pp. 261-265.

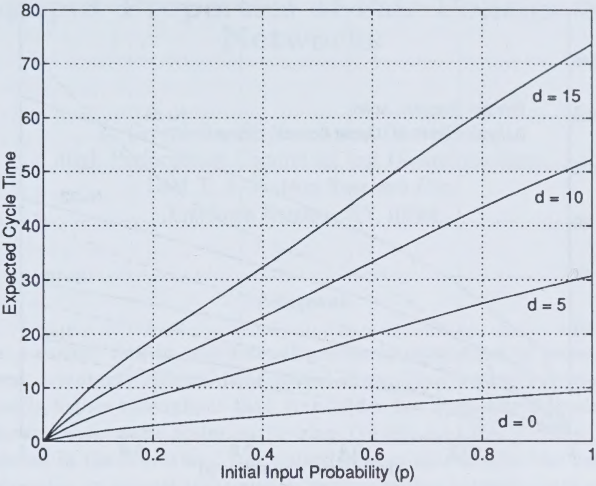


Figure 2: Upper bound for expected cycle time for a 8 input Banyan network using a gated hold protocol with $d = 0,5,10,15$.

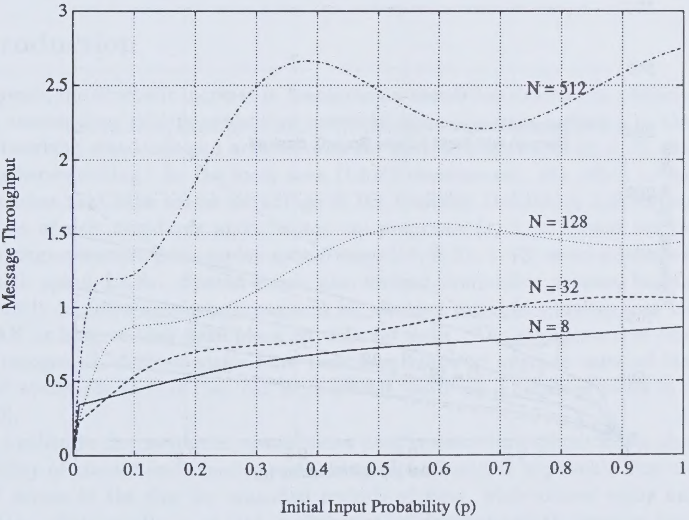


Figure 3: Message throughput for $d = 0$.

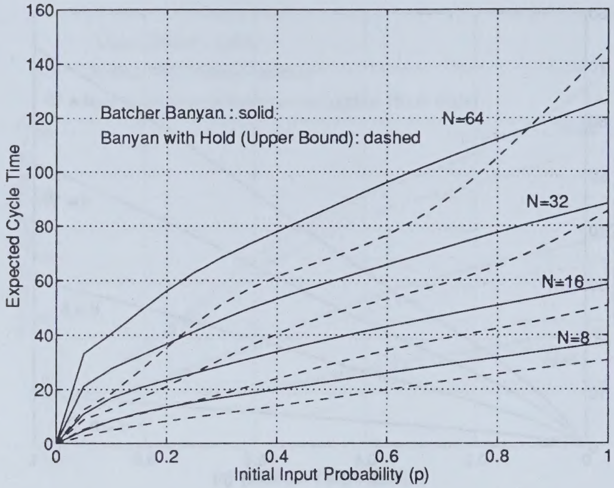


Figure 4: Comparison of upper bound to blocking network with gated hold strategy and simulation/analysis of non-blocking network with gated input strategy for $d = 5$.

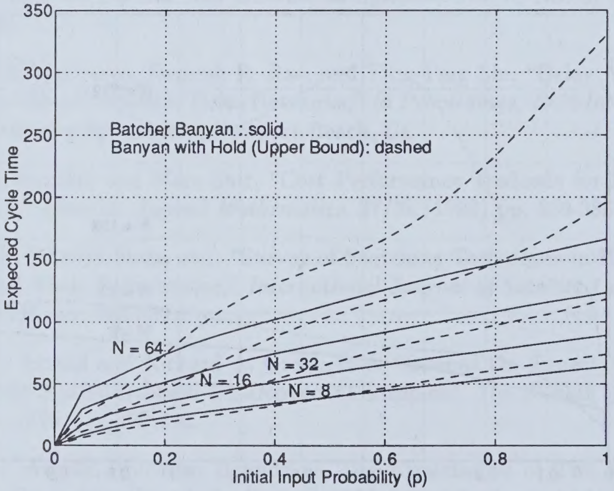


Figure 5: Comparison of upper bound to blocking network with gated hold strategy and simulation/analysis of non-blocking network with gated input strategy for $d = 15$.

Throughput Properties of Fair Policies in Ring Networks

L. Georgiadis, R. Guerin and Israel Cidon[*]
High Performance Computing and Communications
IBM T. J. Watson Research Ctr.
Yorktown Heights, NY 10598

Abstract

We consider a slotted ring in which simultaneous transmissions of messages by different stations is allowed, a property referred to as spatial reuse. Ring networks with spatial reuse can achieve significantly higher throughput than standard token rings but they also introduce the possibility of starvation for some nodes on the ring. To alleviate this problem, various policies have been suggested in the literature. Our objective is to characterize the best possible node throughputs achievable by general transmission policies in ring networks with spatial reuse and then to evaluate the throughput trade-off for a class of policies that has been proposed in the literature in order to avoid starvation. Specifically, we study a policy that is based on the idea of allocating transmission quotas to the nodes. Each node is guaranteed transmission of his quota within a specified interval. We show that by appropriately allocating the quotas, policies that are satisfying general optimality criteria-in particular criteria related to fairness -can be designed. We next study the asymptotic behavior of the quota policy when either the quota sizes or the number of nodes increases.

1 Introduction

In recent years, the dramatic increase in transmission speeds has drastically altered many of the operating assumptions of communication networks. This has been reflected in the emergence of new networking standards and architectures for wide area networks [2, 4, 7], that are based on fast packet-switching. In the local area (LAN) environment, the effort towards defining new approaches that take better advantage of the available technology has similarly resulted in a number of new standards and architecture proposals [1, 3, 9, 8] and led to a renewed interest in rings networks that employ *spatial reuse* [14, 9, 21, 8, 13] as an attractive alternative for new high speed LANs. Spatial reuse, also termed destination release, has the potential to significantly improve network throughput by allowing multiple simultaneous transmissions on the LAN as long as they take place on different links. The significance of this advantage has been recognized, and proposals have even been made to upgrade some of the early high speed LAN standards so that they can support this feature, e.g., erasure nodes in DQDB, etc. [18, 15, 10].

When applied to ring networks, spatial reuse does increase throughput but it also introduces the possibility of *starvation* for nodes on the ring. Specifically, it is possible that certain nodes be denied access to the ring for extended periods of time, while others enjoy uninterrupted transmissions. This results in an unfair allocation of ring bandwidth between the nodes that are sharing the same ring. It is, therefore, necessary to provide mechanisms that will enforce a fair allocation of network resources while preserving their efficient sharing.

The importance of this problem has been recognized, not only in the context of LANs but also for wide area networks, e.g., see [12, 17, 5]. Considerable work has been done to address this issue in rings that employ spatial reuse and several algorithms have been proposed that attempt to preserve fairness without significantly impacting ring throughput [9, 19, 8, 20]. It is beyond the scope of this paper to compare the respective merits of all these algorithms. Rather, our focus is on understanding and sizing the loss of network throughput incurred when attempting to guarantee fairness. This can then be applied to the analysis of existing policies and the design of new and improved ones. Since the focus in this paper is on node throughputs, we assume that the nodes are full with packets always waiting for transmission.

---

[*] and Dept. of Electrical Engineering, Technion, Haifa 32000, Israel

Our objective is to characterize the best possible node throughputs achievable by general transmission policies in ring networks with spatial reuse and then to evaluate the throughput trade-off for a class of policies that has been proposed in the literature in order to avoid starvation. Specifically, we study a policy that is based on the idea of allocating transmission quotas to the nodes, see [9], [8]. Each node is guaranteed transmission of his quota within a specified interval. We show that by appropriately allocating the quotas, policies that are satisfying general optimality criteria-in particular criteria related to fairness -can be designed. We next study the node throughputs as either the quota size or the size of the network increases. It is found that for a fixed number of nodes, as the quota sizes increase, the node throughputs approach the optimal values exponentially fast when there is only one bottleneck node in the ring. When more than one bottleneck nodes exist, the convergence speed is $O(\nu^{-1/2})$, where ν is the quota size. When the number of nodes increases, for a wide class of message destination probabilities the optimal node throughput is achieved even when the quota sizes remain constant.

The structure of the paper reflects the goals and motivations outlined above. Section 2 precisely defines the ring model and the relevant notations. The throughput space of the ring is then characterized in Section 3. Section 4 contains the results on the asymptotic behavior of the policy as either the quota sizes or the number of nodes are increased.

2 System Model

We first introduce some notation. The symbols $\ominus$, $\oplus$ refer respectively to subtraction and addition modulo M, the number of nodes in the ring. If the index k refers to nodes we denote $\sum_{k=i}^{m} x_k := x_i + x_{i\oplus 1} + \ldots + x_{m\ominus 1} + x_m$. We define $\mathcal{M} = \{0, \ldots, M-1\}$ and for a sequence $\{X(k)\}_{k=1}^{\infty}$, we denote $X := X(1)$.

System Model. We consider a unidirectional ring with M nodes. The nodes are numbered starting from zero, so that the node next (downstream) to node i in the direction of message transmission is node $i \oplus 1$. The round trip delay of the ring is 0 (this assumption is made in order to simplify the discussion. As we will see in Section 4 our methodology can also be applied to models with nonzero round trip delays such as the one studied in [16]). The system is slotted, that is, time is divided in slots of length 1 and the nodes can transmit messages at the beginning of each slot. Slot T corresponds to the time interval $[T, T+1)$. We assume that the length of each message is one slot. There is an infinite queue of messages in each node and the destination of the kth message in the queue of node i is denoted by $D_i(k)$. The sequence $\{D_i(k)\}_{k=1}^{\infty}$ consists of i.i.d. random variables independent of the destination sequences in the other nodes. We set $q_{ij} := \Pr(D_i(k) = j)$ (note that we do not exclude the possibility that $q_{ii} > 0$, that is, the message has to travel around the ring - broadcast message). Multiple messages can be transmitted at the same time on the ring. Assume that at the beginning of a slot, node $i \ominus 1$ transmits a message with destination node j. If $j = i$, node i can transmit one of its messages in the same time slot, while it receives the message from node $i \ominus 1$. If $j \neq i$, then, *in the same slot*, node i can either retransmit the upstream message or it can send one of its own messages and store the upstream message for later transmission. Although restrictions are imposed in practice on the number of upstream messages that a node can hold, we consider the operation in this generality initially, in order to describe the throughput space of the system under general scheduling policies (see Section 3).

We now specify the class of admissible message transmission policies. Let $S_{ij}(T)$ be the number of messages sent up to time T, $T \geq 1$, by node i with destination node j and $S_i(T) := \sum_{j=0}^{M-1} S_{ij}(T)$. Let also $R_{ij}(T)$ be the number of messages received by node j with origin node i and $R_i(T) := \sum_{j=0}^{M-1} R_{ij}(T)$. We denote by $\mathbf{\Pi}$ the class of policies that satisfy the following properties.

1. The long term transmission rate of $S_{ij}(T)$, $V_{ij} := \lim_{T\to\infty} S_{ij}(T)/T$, exists for all i, j. Note that we do not require that the limit be the same for all sample paths. That is, V_{ij} can be a random variable.
2. The long term reception rate (throughput) of $R_{ij}(T)$ exists and is equal to the transmission rate V_{ij}.
3. The long term proportion of messages originated at node i with destination node j is equal to the proportion of messages with destination j that are in the queue of node i when the system starts. That is, if $\lim_{T\to\infty} S_i(T) = \infty$, then

$$\lim_{T\to\infty} \frac{S_{ij}(T)}{S_i(T)} = q_{ij}.$$

These assumptions are mainly technical and do not impose any significant restrictions on the class of practical transmission policies. For the system under consideration, however, additional constraints on the admissible policies are imposed. First, the buffer space available at a node for the storage of messages originated in other nodes is limited to a single message. This buffer space is referred to as the "ring buffer." Second, each node provides priority to the messages originated in other stations. Due to these two requirements, a single ring buffer assures the delivery of all messages without the possibility of buffer overflow. It this way, the use of expensive fast buffers is avoided and the fast delivery of messages entering the network is guaranteed. A ring that employs a policy $\pi \in \Pi$ that has the above two additional properties will be called a "buffer insertion ring". For fairness reasons a third constraint is often imposed on the admissible policies. Specifically, it is required that a policy guarantees a finite upper bound on the "channel access time", that is, the time elapsed between two successive transmissions of messages from the same node. The class of policies from Π which have the three additional properties described above will be denoted by Π_0.

3 Throughput Space

Let $V_i = \sum_{j=0}^{M-1} V_{ij}$ be the throughput of node i under a policy $\pi \in \Pi$. We are interested in finding the set of values that the vector $\mathbf{V} := \{V_0, \ldots, V_{M-1}\}$ can take when π is employed. We call this set of values the "throughput space" of the ring and denote it by $\mathcal{V}$.

Assume first that $V_i > 0$, $i \in \mathcal{M}$. Since $\pi \in \Pi$, this implies that $\lim_{T\to\infty} S_i(T) = \lim_{T\to\infty} R_i(T) = \infty$. Let $L_{im}(T)$ be the number of messages with origin node i that passed through node m up to time T ($L_{ii}(T)$ denotes the number of messages with origin node i that were transmitted up to time T). Clearly,

$$\sum_{j=m\oplus 1}^{i\oplus M} R_{ij}(T) \leq L_{im}(T),$$

and since $\sum_{i \in \mathcal{M}} L_{im}(T) \leq T$, we have that

$$\sum_{i \in \mathcal{M}} \sum_{j=m\oplus 1}^{i\oplus M} \frac{R_{ij}(T)}{R_i(T)} \frac{R_i(T)}{T} \leq 1. \tag{1}$$

The properties of π imply that $\lim_{T\to\infty} R_{ij}(T)/R_i(T) = \lim_{T\to\infty} S_{ij}(T)/S_i(T) = q_{ij}$ and therefore, using (1) we have that

$$\sum_{i \in \mathcal{M}} a_{im} V_i \leq 1, \quad m \in \mathcal{M}, \tag{2}$$

where $a_{im} = \sum_{j=m\oplus 1}^{i\oplus M} q_{ij}$; a_{im} is the probability that a message generated by node i is destined to a node that is downstream from node m, in particular, $a_{ii} = 1$. It is easy to see that (2) continues to hold when $V_i = 0$ for some nodes. From the above discussion we see that

$$\mathcal{V} \subset \mathcal{W} := \{\mathrm{v} \in I\!R^M : v_m \geq 0, \sum_{i=0}^{M-1} a_{im} v_i \leq 1, \quad m \in \mathcal{M}\}. \tag{3}$$

It can also be shown (see [11]) that $\mathcal{W}^o \subset \mathcal{V}$, where $\mathcal{W}^o$ is the interior of the set $\mathcal{W}$.

4 Study of a Policy in Π_0

In this and the next sections we will study the following policy.

(π_1^*) As with the randomized policies in Section 3, each node has a ring buffer for storing a single message originated in other nodes. In addition, with node $m \in \mathcal{M}$ there is an associated preassigned integer number $\nu_m \geq 0$ called "quota" and a variable $Q_m(T)$ which is initialized to ν_m, $Q_m(1) = \nu_m$. At the beginning of time T node m performs the following actions.

1. If there is a message in its ring buffer, the node transmits this message. A message transmitted by node $m \ominus 1$ (if any) in the time slot $[T, T+1)$, is stored in the ring buffer of node m.
2. Otherwise,
   (a) if $Q_m(T) > 0$, the node transmits the first message in its queue and sets $Q_m(T) := Q_m(T) - 1$. A message transmitted by node $m \ominus 1$ (if any) in the time slot $[T, T+1)$, is stored in the ring buffer of node m.
   (b) if $Q_m(T) = 0$ the node does not transmit any of the messages in its queue. If node $m \ominus 1$ is transmitting a message with destination a node other than m, node m is retransmitting the same message *in the same slot* (therefore, no message is stored in the ring buffer of node m in this case).

At the time instant T at which $Q_m(T) = 0$ for all $m \in \mathcal{M}$ and the ring buffers of all nodes are empty, we reset $Q_m(T) = \nu_m$. In this case we say that node m is "allocated new quota".

Policy π_1^* is a synchronous version of the algorithm proposed in [8]. The main difference is that in the distributed algorithm of [8], node m resets its quota whenever it receives a circulating token, while under π_1^* all nodes reset $Q_i(T)$ at the same time. The modification is introduced here to simplify the analysis. Also, π_1^* has similarities to the policy considered in [16], which models the operation of the Orwell protocol [9]. The main difference is that in [16] the round trip delay is considered to be one slot, while in our model the round trip delay is zero. Although there are also differences on the order in which the nodes are given permission to transmit on the ring, as will be explained shortly (see the remark after Proposition 1), our results can be directly applied to the model in [16].

The policy described above uses quotas that are integer numbers. We will be interested in the following in policies that allocate quotas according to predetermined ratios. Since any ratios can be expressed either exactly or arbitrarily close with integer numbers, integer quotas are sufficient for most applications. However, if desired, π_1^* can be easily modified to accommodate any ratios (e.g. $\sqrt{2} : \sqrt{3}$). In this case ν_m is interpreted as *average* quota and the policy is implemented as follows. The kth quota allocated to node m is a random variable, $\boldsymbol{\nu}_m(k)$, that takes the values $\lfloor \nu_m \rfloor$ and $\lfloor \nu_m \rfloor + 1$ with probabilities $1 - \nu_m + \lfloor \nu_m \rfloor$ and $\nu_m - \lfloor \nu_m \rfloor$ respectively. The random variables $\boldsymbol{\nu}_m(k)$, $k = 1, \ldots$ are i.i.d and independent of all the processes in the system. It is easy to see that $E\boldsymbol{\nu}_m(k) = \nu_m$. Note that when integer quotas are allocated, $\boldsymbol{\nu}_m(k) \equiv \nu_m$, that is, the modified policy reduces to π_1^*. For the sake of generality, we will study the modified policy in the following and we will also denote it by π_1^*.

From the description of the algorithm we see that under π_1^* the system operates in cycles. At the beginning of cycle k, node $m \in \mathcal{M}$ has quota $\boldsymbol{\nu}_m(k)$ and the end of the cycle, the quotas of all nodes are zero and their ring buffers contain no messages. Formally let $\tilde{B}_m(T)$ be the number of messages in the ring buffer of node m at time T. Let also $T_0 = 1$ and define for $k \geq 0$,

$$\begin{aligned} T_{k+1} &= \min\{T > T_k : \; Q_m(T) = 0, \; \tilde{B}_m(T) = 0, \; m \in \mathcal{M}\}, \\ Q_m(T_{k+1}) &= \boldsymbol{\nu}_m(k+1), \quad m \in \mathcal{M}. \end{aligned}$$

Provided that there are no messages in the ring buffers at time $T = 1$, the sequence $\{T_{k+1} - T_k\}_{k=0}^{\infty}$ consists of i.i.d random variables. Using the terminology in [16] we will call the random variable $T_e := T_1 - T_0$ the "evacuation time" of the ring. As we will see, the evacuation time is crucial in determining the performance of π_1^* in terms of node throughput. We derive next an expression for T_e. Let K_{ij} be the number of messages originated from node i with destination node j during T_e. Let also N_m be the number of messages that are originated or have to be retransmitted by node m during T_e. Clearly,

$$\sum_{j \in \mathcal{M}} K_{ij} = \boldsymbol{\nu}_i$$

and

$$N_m = \sum_{i \in \mathcal{M}} \sum_{j = m \oplus 1}^{i \oplus M} K_{ij}. \tag{4}$$

Since at least one node is transmitting in each slot in the interval $[1, T_e + 1)$ and the quotas are finite, T_e is also finite. Also, since a node can transmit at most one message in each slot, we have

$$T_e \geq U_e := \max_{m \in \mathcal{M}} N_m. \tag{5}$$

The next proposition shows that equality holds in (5).

Proposition 1 *For every M,*

$$T_e = \max_{m \in \mathcal{M}} N_m.$$

Proof. See [11].

Remark. In the model considered in [16], assume that the slot starts from a specific point O on the ring and let τ_e be the first time the slot reaches O and all nodes have completed their quotas. Following essentially the same approach, it can be shown that $\tau_e = \max_{m \in \mathcal{M}} N_m$ and therefore, the evacuation time, $\hat{T}_e$, of the ring studied in [16] can be expressed as

$$\hat{T}_e = \max_{m \in \mathcal{M}} N_m - Y,$$

where $0 \leq Y \leq 1$, is a random variable that expresses the time it takes for the slot to travel from the node that last finished its quota to the point O. Since the asymptotic properties of $\hat{T}_e$ depend again on $\max_{m\in\mathcal{M}} N_m$, our analysis can be applied directly to the model in [16].

Since T_e is bounded, using regenerative arguments it is easy to establish that $\pi_1^* \in \Pi_0$ and that the throughput of node $i \in \mathcal{M}$ is

$$v_i = \frac{E\nu_i}{ET_e} = \frac{\nu_i}{ET_e}. \tag{6}$$

We will investigate the asymptotic properties of this algorithm when either the quota size or the number of nodes becomes large.

4.1 Limiting Behavior of the Algorithm

4.1.1 Node Throughputs for Large Quotas

We consider first the asymptotic behavior of v_i, $i \in \mathcal{M}$, for large quota. Let real numbers $r_i \geq 0$, $i \in \mathcal{M}$, be given. To avoid the trivial case we assume that $r_i > 0$ for at least one $i \in \mathcal{M}$. We set $\nu_i = \nu r_i$, $i \in \mathcal{M}$, $\nu \geq 1$. Whenever needed, to explicitly denote the dependence of a quantity on ν, say quantity X, we write $X(\nu)$. Using (6) and proposition 1 it can be shown that

$$\lim_{\nu\to\infty} v_i(\nu) = \lim_{\nu\to\infty} \frac{\nu r_i}{ET_e(\nu)} = \frac{r_i}{\max_{m\in\mathcal{M}} \left\{\sum_{j\in\mathcal{M}} r_j a_{jm}\right\}}. \tag{7}$$

We now show how (7) can be used in the design of ring access policies. A design objective for such policies is usually associated with an optimization problem in $\mathcal{W}$. For example, in ring networks with spatial reuse, fairness is the main issue. The simplest optimization problem associated with this criterion is to maximize the minimum node throughput in the network

$$\max_{\mathbf{v}\in\mathcal{M}} \min_{m\in\mathcal{M}} v_m. \tag{8}$$

When preferential treatment of the nodes is desirable, the slightly more general problem

$$\max_{\mathbf{v}\in\mathcal{M}} \min_{m\in\mathcal{M}} \frac{v_m}{g_m},\ g_m > 0,\ m \in \mathcal{M}, \tag{9}$$

is appropriate. It is not difficult to see that the vector with coordinates

$$v_i^* = \frac{g_i}{\max_{m\in\mathcal{M}}\{\sum_{j\in\mathcal{M}} a_{jm} g_j\}},\ i \in \mathcal{M}, \tag{10}$$

solves the problem in (9). A stronger optimization problem related to fairness and widely used in the literature is to find the *max-min optimal* vector in $\mathcal{W}$ (see [6, Section 6.5.2] for a description of the fairness properties of max-min optimal vectors). The max-min optimal vector in $\mathcal{W}$ can be easily determined by slightly modifying the algorithm in [6]. Both for problem (8) and the max-min optimization, it is also easy to solve the more general problem obtained by replacing v_m with a reward $f_m(v_m)$, where $f_m(\cdot)$ is a non-decreasing function. Depending on the application, other criteria may be desirable, however, one main characteristic of most of these problems (including those mentioned before) is that the resulting optimal point $\mathbf{v}^*$ lies on the "upper" boundary of $\mathcal{W}$, that is, for some $k \in \mathcal{M}$, the equality $\sum_{j\in\mathcal{M}} a_{jk} v_j^* = 1$ holds. Given such a vector we can pick appropriate quotas so that π_1^* provides node throughputs arbitrarily close to $\mathbf{v}^* = \min\{v_0^* \dots v_{M-1}^*\}$. To see this, let $v^* = \min\{v_i^* : v_i^* > 0,\ i \in \mathcal{M}\}$ and define $r_i = v_i^*/v^*$, $i \in \mathcal{M}$. Then, since $\sum_{j\in\mathcal{M}} a_{jm} r_j v^* \leq 1$, $m \in \mathcal{M}$ and $\sum_{j\in\mathcal{M}} a_{jk} r_j v^* = 1$, we have that

$$v^* = \frac{1}{\max_{m\in\mathcal{M}} \left\{\sum_{j\in\mathcal{M}} r_j a_{jm}\right\}}$$

and therefore,

$$v_i^* = \frac{r_i}{\max_{m\in\mathcal{M}}\left\{\sum_{j\in\mathcal{M}} r_j a_{jm}\right\}}. \tag{11}$$

According to (7), if we pick $r_i = v_i^*/v^*$, as ν becomes large, the vector of node throughputs induced by π_1^* converges to the desired vector $\mathbf{v}^*$. Therefore, once the optimal vector $\mathbf{v}^*$ is determined, it is easy to design the desired policy. For problem (9) as well as the max-min optimization problem, $\mathbf{v}^*$ can be computed very simply, while for other optimization problems, $\mathbf{v}^*$ can be computed by standard numerical procedures. The main disadvantage with this approach is that for the determination of r_i, knowledge of the statistical parameters q_{ij} that determine the performance space is required. As such, the approach can be useful in environments in which these parameters do not change fast. Observe, however, that for problem (9) no such knowledge of statistical parameters is required. Indeed, in this case we can simply pick $r_i = g_i$, $i \in \mathcal{M}$. Once r_i, $i \in \mathcal{M}$, have been determined, the desired optimal point can be approached arbitrarily close by increasing ν. In the next two sections we will examine the dependence of the node throughputs on the quota size.

4.1.2 Bounds on the Rate of Convergence for Large Quota

In this section, we provide bounds on the rate of convergence of policy π_1^* as the quota sizes increase. Since larger quota imply larger channel access times, it is important to know whether the node throughput approaches the optimal throughput quickly as the quota size increases. For lack of space we only provide an outline of the basic arguments. Let m_0 be one of the *bottleneck nodes*, that is,

$$\sum_{j\in\mathcal{M}} r_j a_{jm_0} = \max_{m\in\mathcal{M}}\left\{\sum_{j\in\mathcal{M}} r_j a_{jm}\right\}$$

Observe from (4) that

$$EN_m = \nu \sum_{j\in\mathcal{M}} r_j a_{jm} \tag{12}$$

and therefore, we can write for the limit vector in (7),

$$v_i^* := \frac{r_i}{\max_{m\in\mathcal{M}}\left\{\sum_{j\in\mathcal{M}} r_j a_{jm}\right\}} = \frac{\nu r_i}{\max_{m\in\mathcal{M}}\{EN_m\}},\ i \in \mathcal{M}.$$

On the other hand, the throughput of the system when $\nu_i = r_i\nu$ is

$$v_i(\nu) = \frac{r_i\nu}{E\left(\max_{m\in\mathcal{M}}\{N_m\}\right)}.$$

Therefore, the loss in throughput is due to the fact $E\left(\max_{m\in\mathcal{M}}\{N_m\}\right) \geq \max_{m\in\mathcal{M}}\{EN_m\}$, i.e., the cycles are elongated since the message destinations are random. However, when the quotas are becoming large, the law of large numbers takes effect and as a result, the maximum of N_m, $m \in \mathcal{M}$, is attained at one of the bottleneck nodes most of the time. If there is only one bottleneck node on the ring the maximum will almost always be located at that node, and the effect of randomness will disappear quickly. However, when at least two bottleneck nodes exist, the effect of randomness tends to persist for larger quotas since the maximum may be attained at different bottleneck nodes at different times.

For simplicity in the exposition we assume in this section that both ν and $r_i, i \in \mathcal{M}$, are positive integers and therefore, no randomization of the allocated quota is needed. According to (7) the maximum throughput achievable by node i is given by

$$v_i^* = \frac{r_i}{\max_{m\in\mathcal{M}}\{\sum_{j\in\mathcal{M}} r_j a_{jm}\}}.$$

The main result of this section is the following proposition.

Proposition 2 *When ν and r_i, $i \in \mathcal{M}$ are positive integers,*

$$0 \le \frac{v_i^* - v_i(\nu)}{v_i^*} \le (B-1)O\left(\frac{1}{\nu^{1/2}}\right) + O\left(e^{-c\nu}\right),$$

wher B is the number of bottelneck nodes on the ring.

From proposition 2 we see that the quota size needed to reach within a small percentage of the optimal node throughput can be relatively large if there is more than one bottleneck node in the ring. However there is another factor, namely the number of nodes on the ring, M, that affects favorably the quota size. Note that the maximum channel access time is upper bounded by the sum of the quota allocated to the nodes. Therefore, for rings with small number of nodes, a proportional increase of the allocated quotas may be acceptable. However, for rings with large number of nodes, the same increase will result in much longer channel access times in the worst case. However, as we will see in the next section, for rings with large number of nodes increasing the allocated quotas may be unnecessary, since for a wide family of message destination probabilities, small quota can provide node throughputs close to the optimal.

4.1.3 Limiting Behavior for Large Number of Nodes

We now consider the situation in which the number of nodes, M, becomes large. As in the previous section, when necessary, we use the notation $X(M)$ to make the dependence of a quantity X on M explicit. It was found in [16] that when $\nu_i = 1$, $i \in \mathcal{M}$ and $p_{ij} = 1/M$, $i, j \in \mathcal{M}$, then $\lim_{M\to\infty} M/ET_e(M) = 2$. For uniform message destination probabilities and when all nodes have the same quotas, the maximum throughput of node i is

$$v_i^u(M) = \frac{1}{\sum_{i=0}^{M-1} \frac{M-i}{M}} = \frac{2}{M+1},$$

independent of i. We conclude that

$$\lim_{M\to\infty} \frac{v_i(M)}{v_i^u(M)} = \lim_{M\to\infty} \frac{M+1}{ET_e(M)2} = 1. \tag{13}$$

Therefore, as the number of nodes increases, the throughput of every node approaches the maximal throughput even when the quota size is one slot. We will investigate this property further under general message destination probabilities and arbitrary fixed quota sizes.

According to (7), given r_i, $i \in \mathcal{M}$, the maximal throughput for node i is

$$v_i^* = \frac{r_i}{\max_{m\in\mathcal{M}}\left\{\sum_{i=0}^{M-1} r_i a_{im}\right\}}.$$

In section 4.1.1 we showed that the throughput of node i can be arbitrarily close to v_i^*, however, this can be achieved by increasing the quotas ($\nu_i = r_i\nu$) of all nodes proportionally. We will show next that for a wide class of message destination probabilities, and as the number of nodes increases, there is no need to increase the quotas in order to approach the optimal node throughputs. We provide first an example that shows that the result cannot hold under arbitrary message destination probabilities.

Example 1 Consider a ring with with $M = 3K$, $K > 1$, $\nu_i = 1$, and for $m = 0, \ldots, K-1$, $q_{3m,3m\oplus 1} = q$, $q_{3m,3m\oplus 2} = 1-q$, $q_{3m\oplus 1,3m\oplus 2} = q_{3m\oplus 2,3m\oplus 3} = 1$. In this case the ring contains K identical noninterfering subrings. The evacuation time takes only the values $1, 2$. Since $T_e = 1$ if and only if all nodes with indices $3m$, $m = 0, \ldots, K-1$, transmit to their neighbors, we have $ET_e = 2 - q^K$ and therefore, $\lim_{M\to\infty} v_i(M) = \lim_{K\to\infty}(2-q^K)^{-1} = 1/2$. The maximal node throughputs under the specified quotas, are $v_i^*(M) = 1/(2-q)$, $i \in \mathcal{M}$, and since $0 \le q \le 1$, we see that in this case the node throughput can be reduced by 50 percent relative to the optimal node throughput if the quota remain small. ▮

Assume that we are given a sequence of real numbers r_i, $i = 0, 1, \ldots$, such that

$$\sup_i r_i < \infty.$$

When the number of nodes is M, let policy π_1^* operate with quota $\nu_i = r_i$, $i \in \mathcal{M}$. We will identify a class of message destination probabilities, $q_{ij}(M)$, $i, j \in \mathcal{M}$, $M = 1, 2, \cdots$, for which policy π_1^* induces node throughputs arbitrarily close to the maximal as the number of nodes increases. As in the previous section, the issue is to identify conditions under which $E\left(\max_{m \in \mathcal{M}} N_m(M)\right) \approx \max_{m \in \mathcal{M}} EN_m(M)$. While the strong law of large numbers was in effect in that section, in the current situation the main reason for the above approximation is that under the appropriate condition on the message destination probabilities, the variability of $N_m(M)$ relative to $\max_{m \in \mathcal{M}} EN_m(M)$ is small. The main result of this section is the following proposition.

Proposition 3 *Let* $\sup_i r_i < \infty$ *and assume that when the number of nodes in the ring is* M, *policy* π_1^* *operates with quota* $\nu_i = r_i$, $i \in \mathcal{M}$. *If for some* $\delta > 0$,

$$\lim_{M \to \infty} \frac{M^\delta}{\max_{m \in \mathcal{M}} \left\{\sum_{i=0}^{M-1} r_i a_{im}(M)\right\}} = 0,$$

then

$$\lim_{M \to \infty} \frac{v_i(M)}{v_i^*(M)} = 1.$$

Proof. See [11].

When $q_{ij} = 1/M$, $i \in \mathcal{M}$ and $r_i(M) = 1$, then $\max_{m \in \mathcal{M}} \left(\sum_{i=0}^{M-1} a_{im}(M)\right) = (M+1)/2$ and Proposition 3 holds. This result was shown in [16] using different methods. The question arises whether the assumption on the destination probabilities in Proposition 3 can be weakened. Examples can be given, however, (see [11]) that show that the numerator M^δ cannot be replaced with $(\ln M)^\beta$, $\beta < 1$. Therefore, the condition in the proposition is close to being necessary.

Acknowledgements. We are grateful to Ilan Kessler and Yuval Shavitt for many helpful discussions and comments.

References

[1] FDDI token ring - media access control. ANSI X3.139, American national Standard, November 1986.

[2] Framework for providing additional packet mode bearer services. CCITT recommendation I.122, CCITT Subworking Party XVIII/1-2, 1988.

[3] Distributed queue dual bus (DQDB) subnetwork of a metropolitan area network (MAN). Standard document P802.6/d14, IEEE Project 802.6, July 1990.

[4] Draft - general B-ISDN aspects. Report R 34, CCITT Study Group XVIII, June 1990.

[5] B. R. Bellur and G. H. Sasaki. A SAT-based network access scheme for fairness in high speed networks. In *Proc. 26th Conf. Infor. Sciences & Sys.*, Princeton University, Princeton, NJ, March 18–20 1992.

[6] D. Bertsekas and R. Gallager. *Data Networks.* Prentice Hall, second edition, 1992.

[7] I. Cidon, I. Gopal, P. M. Gopal, R. Guérin, J. Janniello, and M. Kaplan. The plaNET/ORBIT high speed network. Technical report, High Performance Computing and Communications, IBM T. J. Watson Research Center, P.O. Box 704, Yorktown Heights, NY 10598, July 1992. Available from the authors.

[8] I. Cidon and Y. Ofek. Metaring - a full-duplex ring with fairness and spatial reuse. pages 969–981, San Fransisco, CA, June 1990. IEEE INFOCOM.

[9] R. M. Falconer and J. L. Adams. Orwell: A protocol for an integrated service local network. *Br. Telecom Technol. J.*, 3(4):21–35, April 1985.

[10] M. W. Garrett and S.-Q. Li. A study of slot reuse in dual bus multiple access networks. *IEEE J. Select. Areas Commun*, 9(2):248–256, February 1991.

[11] L. Georgiadis, R. Guerin, and I. Cidon. Throughput properties of fair policies in ring networks. Technical Report RC 18293(80185), IBM, September 1992. Submitted for Publication.

[12] E. L. Hahne. Round-robin scheduling for max-min fairness in data networks. *IEEE Journal on Selected Areas in communications*, 9(7):1024–1039, September 1991.

[13] P. Heinzmann, H. R. Müller, D. A. Pitt, and H. R. van As. Buffer-insertion cell-synchronized multiple access (BCMA) on a slotted ring. In *Proc. Second Intl. Conf. Local Commun. Sys.: LAN and PABX*, Palma, Balearic Islands, Spain, June 26–28 1991.

[14] D. E. Huber, W. Steinlin, and P. J. Wild. SILK: An implementation of a buffer insertion ring. *IEEE J. Select. Areas Commun.*, SAC-1(5):766–774, November 1983.

[15] M. J. Karol and R. D. Gitlin. High-performance optical local and metropolitan area networks: Enhancements of FDDI and IEEE 802.6 DQDB. *IEEE J. Select. Areas Commun*, 8(8):1439–1448, October 1990.

[16] I. Kessler and A. Krishna. On the cost of fairness in ring networks. Technical Report RC 17782, IBM T. J. Watson Research Center, March 1992.

[17] A. K. J. Parekh. *A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks.* PhD thesis, Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA 02139, February 1992. No. LIDS-TH-2089.

[18] M. A. Rodrigues. Erasure node: Performance improvements for the IEEE 802.6 MAN. In *Proc. INFOCOM'90*, pages 636–643, San Francisco, CA, June 1990.

[19] H. J. F. Sabourin. Offset flow protocol. In *Proc. Second IEEE Workshop Future Trends Distrib. Comput. Sys.*, pages 185–189, 1990.

[20] R. Simha and Y. Ofek. Starvation-free access protocol for a full-duplex buffer insertion ring local area network. *Comput. Networks ISDN Syst.*, 21(2):109–120, April 1990.

[21] T. Takeda and K. Tanno. Register insertion/self-token protocol for high speed ring LANs. In *Proc. 15th Conference on Local Computer Networks*, pages 60–64, 1990.

Access Control to Two Multiserver Loss Queues in Series

Cheng-Yuan Ku and Scott Jordan
cooperku@eecs.nwu.edu and scott@eecs.nwu.edu
Department of Electrical Engineering
Northwestern University, Evanston, IL 60208-3118

Abstract

We consider admission policies to two multiserver loss queues in series with two types of traffic. The first type requires service at each of the queues; the second type requires service at only the 2nd queue. We show that under appropriate conditions the optimal admission policy is given by a switching curve. We characterize the form of this curve and its variation with system parameters.

1. Introduction

The merging of telephone and computer networks is introducing multiple resources into networks, and information is becoming increasingly distributed across the networks. Related services are being integrated onto a single network rather than being offered on separate uncoordinated networks.

Resource allocation, including access control, is becoming a more important issue in these networks. As the speed of the transport and processing increases, we believe multiserver facilities with little capability to buffer bursty services will become more common. (See [1] for a simultaneous resource allocation model)

In this paper, we study two multiserver loss queues in series. By characterizing optimal admission policies, we hope to gain some understanding into reservation of resources in integrated service networks. The queueing model, however, is quite general, and applies to situations outside networking.

2. A Model

In this section we present a sequential resource allocation (SRA) model, and discuss related research.

Suppose a service type 1 requires use of a resource A and then when it is finished with A it requires use of a resource B. Suppose a service type 2 only requires use of a resource B. If there are many As and Bs, how should we manage these resources?

This work was supported by the National Science Foundation under grant NCR-9109223, and by Ameritech under an Ameritech Research Fellowship.

This system can be described as follows:

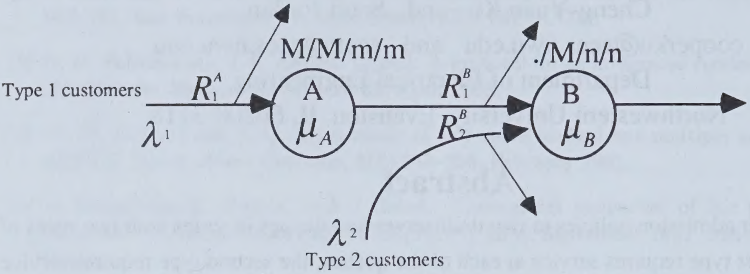


Fig-1 SRA model for continuous time

Station A : M/M/m/m queueing system
λ_1 : Poisson arrival rate of type 1 customers going to station A
μ_A : Exponential service rate of servers at station A
R_1^A : Revenue for service to type 1 customer at station A
Station B : ./M/n/n queueing system
λ_2 : Poisson arrival rate of type 2 customers going into station B
μ_B : Exponential service rate of servers at station B
R_1^B : Revenue for service to type 1 customer at station B
R_2^B : Revenue for service to type 2 customer at station B
Assumption 1 : $R_1^B > R_2^B$: We prefer type 1 customers to type 2 customers at station B.
Assumption 2 : Revenue is collected when a customer enters service.

Many papers have addressed optimal control of admission to a single queue. (See [2] for a good survey.) Naor [3] studied critical-number policies in steady state in a M/M/1 queue. This approach has been generalized to a M/D/1 queue [4], a M/M/c queue with state -dependent benefit [5], a general birth-death congestion system [6], a GI/M/c queue [7] and a general input-output service facility [8].

A few papers have addressed optimal control of admission to multiple queues. Davis [9] studied two exponential servers in parallel with separate queues. Stidham et. al. [10,12] considered exponential single servers in series. Hajek [11] worked on a system of two interacting stations. All of these studies assumed infinite queues. In this paper, we focus on loss queues. We use dynamic programming techniques, similar to previous papers, but some additional complexity is presented by boundary states due to the finite buffers.

For our system, if a service at station A is processing, we know there will be one respective request at station B in the near future. So, the number of customers at station A can be regarded as an amount of reservation for type B resources. But now, a question arises. What is the optimal admission policy? This system can be modeled as a two dimensional Markov chain. State (i,j) is number of customers at stations A and B respectively. When in state (i,j), the system manager should make a decision to accept or deny possibly oncoming customers in order to maximize expected revenue. This is a Markov decision process, and we use the value iteration approach from dynamic programming to answer this question. Previous research often formulated their system as a semi-Markov decision process in which the system is observed at the epochs of arrivals. We find it more convenient and equivalent to approximate our continuous time model by a discrete time model.

The exponential interarrival times and service times are approximated by geometric distributions. We set the length of a discrete time unit to be T. We equate the probability of any one event occurring during a length of time T in the continuous and discrete time systems (within o(T)), and we assume the probability of two or more events at a single time in the discrete system is zero.

The original system can therefore be approximated as follows:

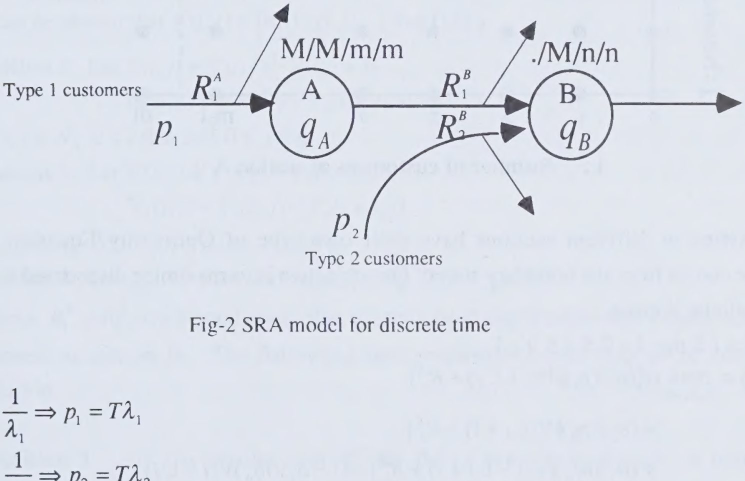


Fig-2 SRA model for discrete time

$$\frac{T}{p_1} = \frac{1}{\lambda_1} \Rightarrow p_1 = T\lambda_1$$

$$\frac{T}{p_2} = \frac{1}{\lambda_2} \Rightarrow p_2 = T\lambda_2$$

$$\frac{T}{q_A} = \frac{1}{\mu_A} \Rightarrow q_A = T\mu_A$$

$$\frac{T}{q_B} = \frac{1}{\mu_B} \Rightarrow q_B = T\mu_B$$

p_1: probability of success in one period for geometric arrival process of type 1 customers

p_2 :probability of success in one period for geometric arrival process of type 2 customers

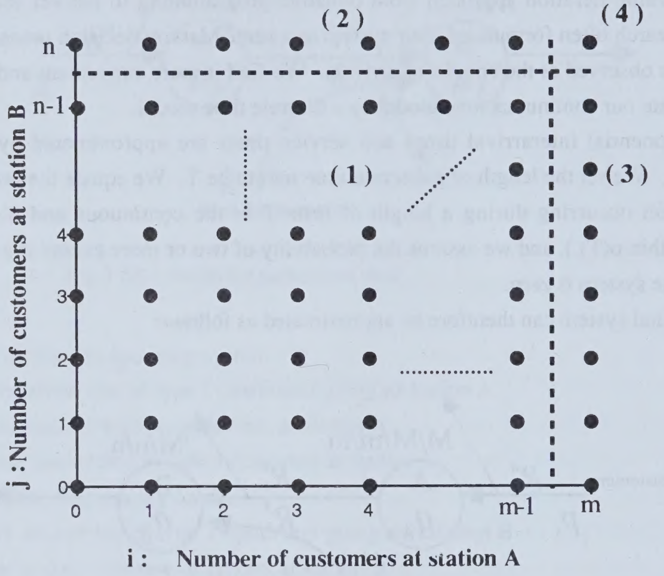
q_A: probability of success in one period for geometric service process at station A

q_B : probability of success in one period for geometric service process at station B

3. Access control

In this section, we present a series of propositions to characterize the optimal admission policy. Due to space limitation, only sketches of the proofs are presented.

On the basis of this model, we can write the dynamic programming Optimality Equations. The states will be divided into four sections as follows.


(2)
(4)
(1)
(3)
n
n-1
4
3
2
1
0
j :Number of customers at station B
0
1
2
3
4
m-1
m
i : Number of customers at station A

The states in different sections have their own type of Optimality Equation. The difference comes from the boundary states. Our objective is to maximize discounted revenue over an infinite horizon.

(1) For $0 \le i \le m-1$, $0 \le j \le n-1$,

$$\begin{aligned} V(i,j) = \max_{a_1,a_2,a_3} \alpha\{&(a_1)(p_1)[V(i+1,j)+R_1^A] \\ &+(a_2)(p_2)[V(i,j+1)+R_2^B] \\ &+(a_3)(iq_A)[V(i-1,j+1)+R_1^B]+(1-a_3)(iq_A)V(i-1,j) \\ &+(jq_B)V(i,j-1) \\ &+(1-(a_1)p_1-(a_2)p_2-iq_A-jq_B)V(i,j)\} \end{aligned}$$

(2) For $0 \le i \le m-1$, $j = n$,

$$\begin{aligned} V(i,j) = \max_{a_1} \alpha\{&(a_1)(p_1)[V(i+1,j)+R_1^A] \\ &+(iq_A)V(i-1,j) \\ &+(jq_B)V(i,j-1) \\ &+(1-(a_1)p_1-iq_A-jq_B)V(i,j)\} \end{aligned}$$

(3) For $i = m$, $0 \le j \le n-1$,

$$\begin{aligned} V(i,j) = \max_{a_2,a_3} \alpha\{ &(a_2)(p_2)[V(i,j+1)+R_2^B] \\ &+ (a_3)(iq_A)[V(i-1,j+1)+R_1^B] + (1-a_3)(iq_A)V(i-1,j) \\ &+ (jq_B)V(i,j-1) \\ &+ (1-(a_2)p_2 - iq_A - jq_B)V(i,j)\} \end{aligned}$$

(4) For $i = m$, $j = n$,

$$\begin{aligned} V(i,j) = \alpha\{ &(iq_A)V(i-1,j) \\ &+ (jq_B)V(i,j-1) \\ &+ (1 - iq_A - jq_B)V(i,j)\} \end{aligned}$$

The admission control is determined by the optimal values for the decision vector $a = (a_1, a_2, a_3)$. Each decision parameter is a function $(i,j) \to \{0,1\}$.

a_1=1 iff admit customer type 1 at station A.

a_2=1 iff admit customer type 2 at station B.

a_3=1 iff admit customer type 1 at station B.

Definition 1 Let $V_0(i,j)$ be an arbitrary bounded function, and for $h \in N, h \neq 0$, define

$$V_h(i,j) = \max_a \{\alpha \sum_k \sum_l P_{(i,j)\to(k,l)}(a)[R((i,j),(k,l)) + V_{h-1}(k,l)]\}$$

where $P_{(i,j)\to(k,l)}(a)$ is one - step transition probability and $R((i,j),(k,l))$ is the revenue. It also can be shown that $V(i,j) = \lim_{h\to\infty} V_h(i,j)$. (See [13])

Definition 2 Let $\Delta(i,j) \equiv V(i,j) - V(i,j+1)$

$$\Delta_h(i,j) \equiv V_h(i,j) - V_h(i,j+1)$$

for $h,i,j \in N$, $0 \le i \le m$ and $0 \le j \le n-1$

Definition 3 Let $\nabla(i,j) \equiv V(i,j) - V(i+1,j)$

$$\nabla_h(i,j) \equiv V_h(i,j) - V_h(i+1,j)$$

for $h,i,j \in N$, $0 \le i \le m-1$ and $0 \le j \le n$

Definition 4 Let $j_p(i,h)$ such that $\Delta_h(i,j) > R_2^B$ iff $j \ge j_p(i,h)$

Since $R_1^B > R_2^B$, it's a good guess that we only need apply admission control upon type 2 customers at station B. The following two propositions will help us to come to this conclusion.

Proposition 1 $\Delta(i,j)$ is smaller than R_1^B for $0 \le i \le m$ and $0 \le j \le n-1$. It is optimal to always admit type 1 customers at station B. That is, $a_3 = 1$.

< Proof >

Base step : Choose initial values to let $\Delta_0(i,j) < R_1^B$, $\nabla_0(i,j) < R_1^A$ and be monotonically increasing on i for fixed j and monotonically increasing on j for fixed i.

Induction step : Assume $\Delta_h(i,j) < R_1^B$. After using complicated algebra we can prove it's also true for $\Delta_{h+1}(i,j)$.

Proposition 2 $\nabla(i,j)$ is smaller than R_1^A for $0 \le i \le m-1$ and $0 \le j \le n$. It is optimal to always admit type 1 customers at station A. That is, $a_1 = 1$.

< Proof > Similar to proposition 1.

Proposition 3 $\Delta(i,j)$ is monotonically increasing on j for fixed i. The optimal policy is a threshold policy.

< Proof >

Base step : Choose the initial values such that $\Delta_0(i,j) < R_1^B$, $\nabla_0(i,j) < R_1^A$ and it is monotonically increasing on j for fixed i.

Induction step : Assume $\Delta_h(i,j)$ is monotonically increasing. After using complicated algebra, we can prove it's also true for $\Delta_{h+1}(i,j)$.

However optimal policy, for specific column i, rejects type 2 customers while $\Delta(i,j) > R_2^B$, so there is a threshold in this column. It is the same for the other columns, therefore the optimal policy is threshold.

Proposition 4 $\Delta(i,j)$ is monotonically increasing on i for fixed j. Let $j_p(i) \equiv \lim_{h\to\infty} j_p(i,h)$, then $j_p(i)$ is a decreasing function on i.

< Proof >

Base step : Choose initial values to let $\Delta_0(i,j) < R_1^B$, $\nabla_0(i,j) < R_1^A$ and be monotonically increasing on i for fixed j and monotonically increasing on j for fixed i. From proposition 3, we know that $\Delta_h(i,j)$ is monotonically increasing on j for fixed i for any h.

Induction step : Assume $\Delta_h(i,j)$ is monotonically increasing on i for fixed j. After using complicated algebra we can prove it's also true for $\Delta_{h+1}(i,j)$.

We know the optimal policy is threshold curve. Therefore, $j_p(i)$ is a decreasing function on i.

These four propositions characterize optimal admission control to the pair of queues, when type 1 customers are more valuable than type 2. Type 1 customers are always admitted. Type 2 customers are admitted only if the number of free servers at station B exceeds some threshold. Furthermore, the number of servers reserved at station B for further arrivals of type 1 increases with the number of type 1 customers at station A. We further investigate the variation of this threshold with changes in system parameters in the next section.

4. Experimental Results

Experimental results were obtained by the method of successive approximation. We find the optimal value function for all states, and from that infer the optimal policy. For every experiment, we show the threshold switching curve for the optimal policy. During the successive approximation, standard substitution method is employed. That is, we refresh all the expected rewards simultaneously in every recursive loop. Furthermore, the norm-2 criterion is selected to measure the precision of approximation. The following are ten experiments. They are chosen to show some interesting phenomenon.

(1)Arrival rate of stage A ? 1

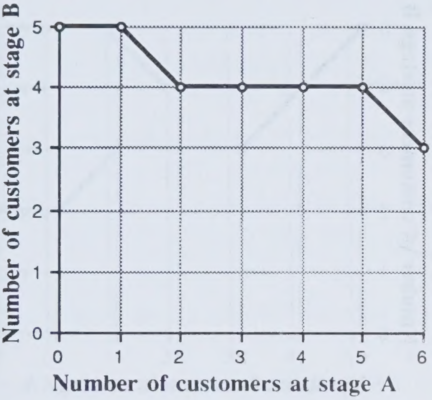
Arrival rate of stage B ? 0.8

Service rate of stage A ? 1.5

Service rate of stage B ? 3

Discount factor ? 0.99999

R_1^A ? 100 R_1^B ? 300 R_2^B ? 50



(2)Arrival rate of stage A ? 1

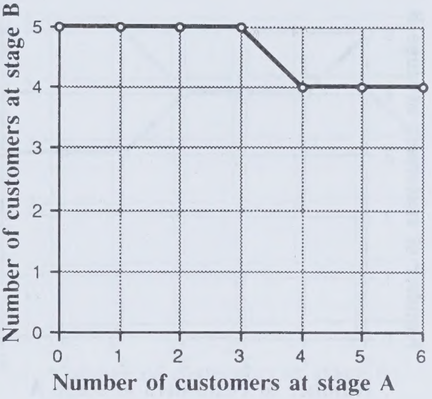
Arrival rate of stage B ? 0.8

Service rate of stage A ? 1.5

Service rate of stage B ? 3

Discount factor ? 0.9999

R_1^A ? 100 R_1^B ? 300 R_2^B ? 50



(3)Arrival rate of stage A ? 1

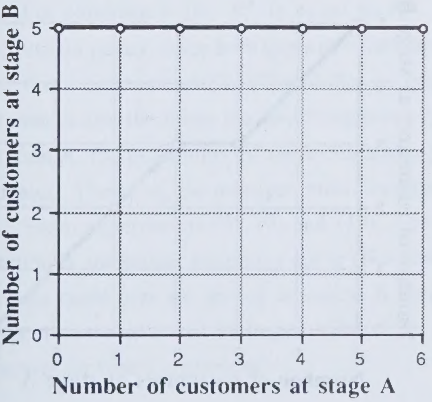
Arrival rate of stage B ? 0.8

Service rate of stage A ? 1.5

Service rate of stage B ? 3

Discount factor ? 0.99

R_1^A ? 100 R_1^B ? 300 R_2^B ? 50



(4)Arrival rate of stage A ? 1

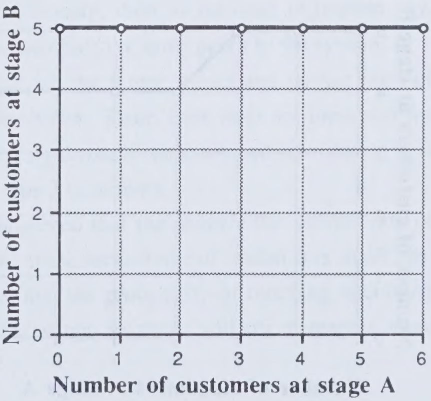
Arrival rate of stage B ? 0.8

Service rate of stage A ? 1.5

Service rate of stage B ? 3

Discount factor ? 0.9999

R_1^A ? 100 R_1^B ? 300 R_2^B ? 300



(5)Arrival rate of stage A ? 1

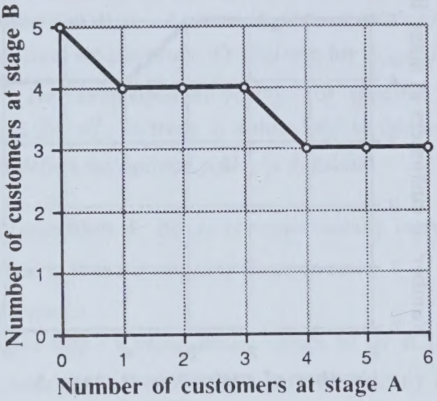
Arrival rate of stage B ? 0.8

Service rate of stage A ? 1.5

Service rate of stage B ? 3

Discount factor ? 0.9999

R_1^A ? 100 R_1^B ? 300 R_2^B ? 10



(6)Arrival rate of stage A ? 1

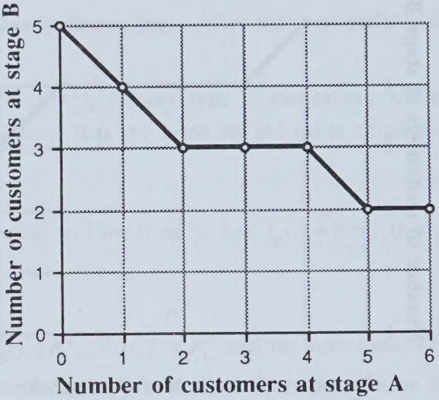
Arrival rate of stage B ? 0.8

Service rate of stage A ? 1.5

Service rate of stage B ? 3

Discount factor ? 0.9999

R_1^A ? 100 R_1^B ? 300 R_2^B ? 2



(7)Arrival rate of stage A ? 3

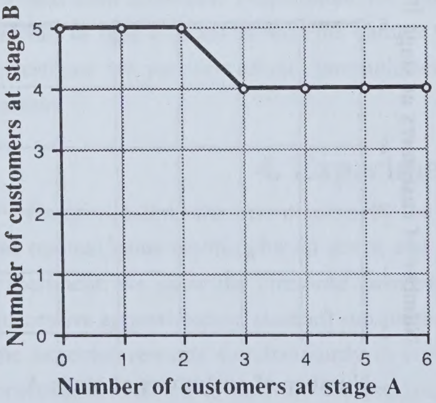
Arrival rate of stage B ? 0.8

Service rate of stage A ? 1.5

Service rate of stage B ? 3

Discount factor ? 0.9999

R_1^A ? 100 R_1^B ? 300 R_2^B ? 50



(8)Arrival rate of stage A ? 10

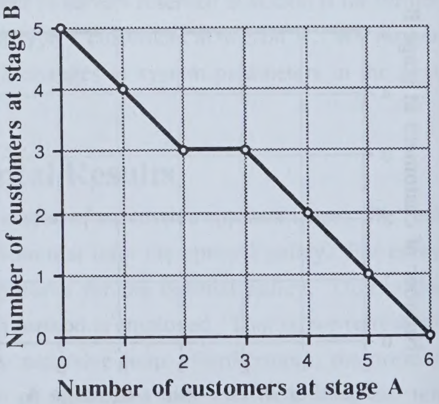
Arrival rate of stage B ? 0.8

Service rate of stage A ? 10

Service rate of stage B ? 3

Discount factor ? 0.9999

R_1^A ? 100 R_1^B ? 300 R_2^B ? 50



(9)Arrival rate of stage A ? 1

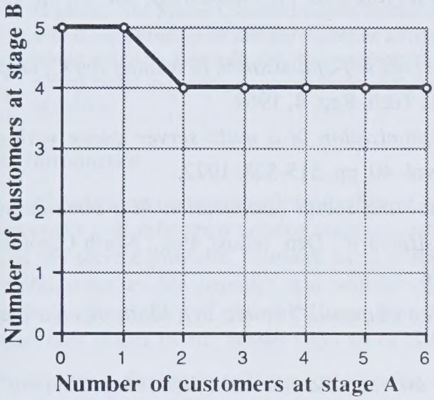
Arrival rate of stage B ? 0.8

Service rate of stage A ? 1.5

Service rate of stage B ? 0.7

Discount factor ? 0.9999

R_1^A ? 100 R_1^B ? 300 R_2^B ? 50



(10)Arrival rate of stage A ? 1

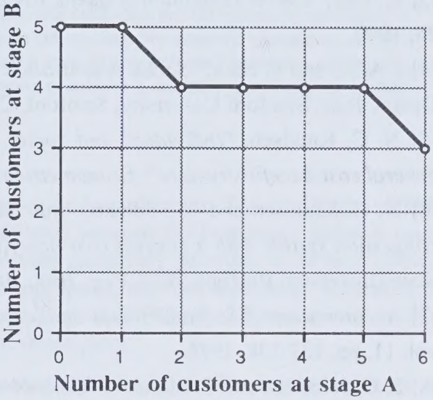
Arrival rate of stage B ? 0.8

Service rate of stage A ? 1.5

Service rate of stage B ? 0.1

Discount factor ? 0.9999

R_1^A ? 100 R_1^B ? 300 R_2^B ? 50



From experiments (1), (2) and (3), we find out the smaller the discount factor is, the less expected revenues are. A smaller discount factor will make the current reward more important. So, the system manager tends to accept type 2 customers and the switching curve goes higher.

From experiments (2), (4), (5), and (6), it can be observed that the larger difference between R_1^B and R_2^B is, the faster the policy curve goes down. That's because the system manager tends to reject type 2 customers and reserve more empty spaces for oncoming type 1 customers.

For experiment (4), R_1^B is equal to R_2^B. Obviously, there is no need to impose any admission policy, since both types of customers contribute the same profit to the system.

From experiments (2), (7) and (8), we notice that the larger arrival and service rates of station A are, the faster the switching curve goes down. Since, both rates are increased for station A, the probability of more customers getting through station A and into station B is greater. Therefore, the manager tends to refuse type 2 customers.

From experiments (2), (9) and (10), it is observed that the smaller the service rate of station B, the sooner the policy curve goes down. If the service rate of station B is small, that means customers are served at station B longer and the probability of blocking oncoming type 1 customers will be larger. Therefore, the system manager will try to reserve more servers for type 1 customers.

References

[1] S. Jordan and P. P. Varaiya, *"Throughput in Multiple Service Multiple Resource Communication Networks", IEEE Trans. Comm.*, vol. 39, no. 8, pp. 1216-1222, 1991.

[2] S. Stidham, *"Optimal control of admission to a queueing system", IEEE Trans. Automat. Contr.*, vol. AC-30, no. 8, pp. 705-713, 1985.

[3] P. Naor, *"On the regulation of queue size by levying tolls", Econometrica*, vol. 37, pp. 15-24, 1969.

[4] I. Adler and P. Naor, *"Social optimization versus self-optimization in waiting lines"*, Dep. Operat. Res., Stanford University, Stanford, CA, Tech. Rep. 4, 1969.

[5] N. C. Knudsen, *"Individual and social optimization in a multi-server queue with a general cost-benefit structure", Econometrica*, vol. 40, pp. 515-528, 1972.

[6] N. C. Knudsen and S. Stidham, *"Individual and social optimization in a birth-death congestion system with a general cost-benefit structure"*, Dep. Indust. Eng., North Carolina State University, Raleigh, Tech. Rep. 76-8, 1976.

[7] A. Simonovits, *"Self and social optimization in queues", Studia Sci. Math. Hungarica*, vol. 11, pp. 131-138, 1976.

[8] S. G. Johansen and S. Stidham, *"Control of arrivals to a stochastic input-output system", Adv. appl. Prob.*, vol. 12, pp. 972-999, 1980.

[9] E. Davis, *"Optimal control of arrivals to a two-server queueing-system with separate queues"*, Ph.D. dissertation, Program Operat. Res., North Carolina State University, Raleigh, 1977.

[10] H. Ghoneim and S. Stidham, *"Optimal control of arrivals to two queues in series", Europe. J. Operat. Res.*, 1985.

[11] B. Hajek, *"Optimal control of two interacting service stations", IEEE Trans. Automat. Contr.*, vol. AC-29, pp. 491-499, 1984.

[12] S. Stidham and R. Weber, *"Control of service rates in cycles and series of queues"*, Stat. Lab., Univ. Cambridge, Cambridge, 1983.

[13] Sheldon M. Ross, *Introduction to Stochastic Dynamic Programming*, Academic Press, 1983.

Strict Peak Rate Enforcement at the ATM Network Entry Point

Jack Brassil

AT&T Bell Laboratories
Murray Hill, New Jersey 07974

Policing the peak cell rate of a connection is a required traffic regulation function performed in an Asynchronous Transfer Mode network. Boyer, et. al. have proposed a peak cell rate policer and traffic shaper, known as the *Spacer-Controller*, to perform Usage Parameter Control and cell spacing functions. The Spacer-Controller passes, delays, or discards cells, such that the output stream complies with the negotiated peak rate and maintains a minimum cell spacing. We study the performance of this device acting on a periodic cell stream subject to delay jitter. We analyze and simulate a general cell arrival model to investigate proper device dimensioning to achieve a desired communications quality of service.

1. Introduction

Maintaining communications quality of service in Asynchronous Transfer Mode (ATM) networks will rely upon service contracts established between the broadband network users and the service provider. Network users will provide parametric descriptions of their offered traffic to the service provider, who will allocate appropriate network resources. Thereafter, it is the responsibility of network Usage Parameter Control (UPC) mechanisms to ensure that each user abides by the agreed upon set of traffic descriptors.

Policing the peak cell rate of a traffic source is a crucial traffic regulation function performed on an ATM network connection [1]. Among the traffic descriptors a source may negotiate with a network, the peak cell rate is relatively simple for the user to estimate and the service provider to police. Policing connections is necessary for several reasons. First, it is required to discourage malicious noncompliant users, those purposely exceeding their negotiated rate. Second, it is required to maintain the existing service contracts of compliant users in the event of customer equipment failure. In general, the network resources (e.g., output buffers in an ATM switch) required to maintain service quality increase if sources are permitted to exceed their negotiated peak rates. Hence, efficient use of network resources and the maintenance of service contracts demand the policing of connections.

Traffic shaping is a second regulation function optionally performed on an ATM connection. Shaping, or adjusting the time structure of a cell stream, can reduce cell delay variations (CDV) introduced by upstream network elements (e.g., multiplexors, switches). Traffic shaping can potentially facilitate switch dimensioning and, in certain cases, help utilize network resources efficiently [2, 3].

Policing and shaping an ATM connection can be appropriate at various points within public or private ATM networks. For example, a traffic source may be connected to the public broadband network through a private ATM access network found on a customer premises (e.g., an ATM local area network). Cell delay variations can be introduced in multiplexing stages in the private network, altering the structure of the cell stream arriving to the public network. Prior to exiting the customer premises, a cell stream might be policed and shaped to ensure compliance with its negotiated traffic descriptor (e.g., peak cell rate). Similar functions might also be performed by the Usage or Network Parameter Control (UPC/NPC) mechanisms in the public network. Here, however, the purpose of the regulation might be different. For example, peak rate policing actions may be taken to protect the service contracts of other public network users, while traffic shaping might be used to smooth bursty traffic to increase the utilization of downstream switches.

In this paper we consider an approach to peak rate policing and traffic shaping a jittered cell stream on a virtual channel connection (VC). Boyer, et. al. [4, 5, 6] have proposed a peak rate policer and cell spacer known as the *Spacer-Controller*. The Spacer-Controller (SC) passes, delays, or discards cells associated with a cell stream, such that the output stream complies with a negotiated peak rate, and maintains a minimum cell spacing. The SC is designed to pass or delay compliant traffic, while discarding noncompliant traffic. The SC uses a simple, single parameter threshold mechanism to determine the tradeoff between cell delay and cell discard.

We focus on a simple, idealized source which continuously emits cells with a constant period. Though we consider virtual channels, the mechanisms we discuss also apply to regulating traffic on Virtual Paths (VPs). Much of the work we describe here also applies to peak rate policing of bursty or variable bit rate (VBR) traffic. However, it is likely that shaping VBR traffic will involve a larger collection of traffic parameters, which will lead to a more complicated enforcement procedure than we describe here. For example, Elwalid and Mitra [7] have studied the performance of an *access regulator* capable of both policing and shaping bursty traffic. In Section 6 we will mention an approach to regulating a more general arrival stream than we consider here.

The remainder of this paper is organized as follows. In the next section we review the definition of the peak cell rate traffic descriptor, and summarize the *equivalent terminal* model of a traffic source. We then introduce the operation of the Spacer-Controller in some detail in Section 3. In Section 4 we consider its performance, focusing on device dimensioning and the tradeoff between cell discard and delay. Select simulation results are presented in Section 5 to verify and extend the performance analysis. In the final section we summarize our results and discuss a number of open problems requiring further research.

2. Peak Rate Definition and the Equivalent Terminal Model

Both CCITT and the ATM Forum have defined the peak cell rate of an ATM connection [8, 9] based on an *equivalent terminal* model of a traffic source. Under the definition, peak cell rate is determined by the logical minimum spacing between cells generated at a traffic source, rather than the actual spacing of cells arriving to a monitoring or policing point[1]. This definition and terminal model promises to facilitate peak rate monitoring and enforcement. In addition to specifying the peak rate of a connection, users are required to specify a second traffic parameter, the Cell Delay Variation (CDV) tolerance. CDV tolerance is intended to provide a safety margin against delay jitter. For example, suppose a cell stream is subject to delay jitter in traversing a customer premises ATM network, prior to entering a public ATM network. CDV tolerance provides a safety margin, so that the cell stream remains compliant with the negotiated peak cell rate upon entering the public network (i.e., at the UPC). No approach has been recommended for setting the value of the CDV tolerance traffic parameter, however it may be chosen to correspond to the maximum possible induced delay jitter, if such a value exists and is known.

---

[1] The peak cell rate of an ATM connection is the inverse of the minimum interarrival time between requests to send an ATM Protocol Data Unit at the Physical Layer Service Access Point of the equivalent terminal. These interarrival times within the equivalent terminal are a model, and do not correspond to the actual spacing of cells emitted by the source.

In the remainder of this paper we consider a single periodic source which emits cells with period T. Cells on the connection are subject to delay jitter, and we assume they may be discarded (or tagged). During or prior to call initialization the source negotiates a peak rate $1/\hat{T}$ and CDV tolerance τ, both of which we allow to be arbitrary. Our objective is to determine the resulting QoS when the connection is subject to policing and shaping by a Spacer-Controller, which we describe in the following section.

3. Operation of the Spacer-Controller

Suppose cells associated with a virtual channel are to be regulated by a Spacer-Controller (SC). In general, the traffic stream arriving to the regulation point will be a multiplexed collection of VCs. In this case, acting on an individual VC requires that the multiplexed streams be demultiplexed prior to the Spacer-Controller, and subsequently remultiplexed. Remultiplexing unfortunately presents an opportunity for cell delay variations to alter the regulated traffic streams, since multiple cells might contend for a single slot at the output multiplexor. Hence, if multiplexed streams are to be shaped, each SC should be properly dimensioned to compensate for this multiplexor jitter. For now, however, we will introduce the operation of the SC as if the arriving traffic stream did not require demultiplexing (i.e., a single VC on the trunk). We will then mention how to compensate for jitter introduced by downstream remultiplexing.

The SC implements a virtual scheduling algorithm for each enforced VC, as depicted in the flowchart of Figure 1. The algorithm is implemented with a small number of variables and counters, which we now describe. Consider the cells arriving to an SC on an active VC, and let the duration of a cell slot equal one unit of time. The "current" cell is taken to arrive at time t. A target departure time for the current cell is called the Theoretical Re-emission Time (*TRT*). The Last Re-emission Time (*LRT*) identifies the time of the most recent cell departure from the SC. Since cells may be discarded or queued by the SC, the most recent cell departure need not correspond to the most recent cell arrival. The SC ensures that the current cell will not be emitted prior to its Theoretical Re-emission Time; its value is calculated so that the current cell will not be emitted before the minimum cell spacing interval is reached. That is, if the current cell departs after time *TRT*, at least $\lceil \hat{T} \rceil$ cell slots have passed since the last cell departure at time *LRT*.

The Spacer-Controller shapes the cell stream by either passing, delaying or discarding arriving cells. The action taken on the current cell is determined by the relative values of t, *TRT*, and the constant τ, as follows:

1. $t \geq TRT$
   Description: The current cell arrives after its earliest scheduled departure.
   Action: The SC passes the current cell immediately (i.e., takes no action). The value of *TRT* is reset to t.
2. $TRT - \tau \leq t < TRT$
   Description: The current cell arrives no more than τ time units prior to its earliest scheduled departure.
   Action: The SC buffers and delays the current cell, not emitting it until time *TRT*.
3. $t < TRT - \tau$
   Description: The current cell arrives more than τ time units prior to its earliest scheduled departure.

Action: The SC discards the current cell.

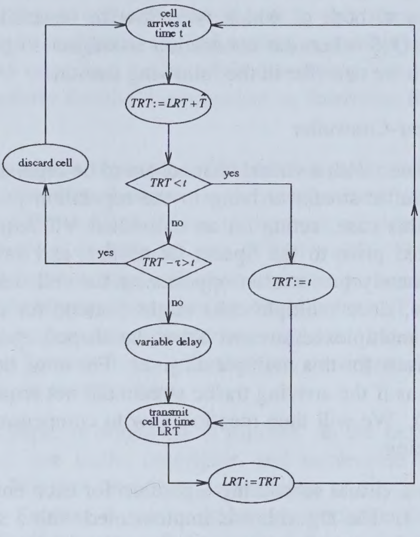


Figure 1 - The algorithm implemented by the Spacer-Controller.

Boyer proposes calculating the *TRT* in two distinct ways. In the first approach, called *TRT-spacing*, delay jitter potentially introduced by the downstream multiplexor is essentially ignored. The value of *LRT* is not taken to be the actual time of the last departure (from the multiplexor), but rather the maximum of 1) the TRT for the last departing cell, and 2) and the arrival time of the last departing cell. Alternatively, Boyer proposed *ART-spacing*, which uses the actual departure time of the last cell (from the multiplexor), called the Actual Re-emission Time (ART). In this paper we study the simpler *TRT-spacing*, recognizing that the SC can be dimensioned with additional margin to compensate for CDV introduced by remultiplexing.

We now pause to make several simple observations on the operation of the SC. The cell stream departing an SC clearly satisfies the minimum cell spacing interval $\hat{T}$. As desired, the SC does not alter the time structure of an arriving stream satisying this minimum spacing. Cells arriving "much too early" to the SC are discarded while those arriving "slightly early" are buffered and delayed. Cells arriving "on time" or "late" are passed immediately. The dimensioning parameter τ and a source's selection of negotiated minimum spacing $\hat{T}$ determine the tradeoff between cell loss and delay. The maximum delay of a nondiscarded cell in the SC is $\lceil \tau \rceil$ time units, and the maximum number of cells which must be buffered is $1 + \lfloor \tau / \hat{T} \rfloor$. With the value of $\hat{T}$ fixed, increasing the value of τ lowers the cell discard probability. In Figure 2 we illustrate the actions taken by the SC for a particular arrival sequence.

At this point it is helpful to introduce some notation and relationships between SC operating variables. To ease presentation we will assume that the variables τ, $\hat{T}$, and T are positive integers. Let cell j, $j=0,1,2,\cdots$, be the $j+1^{st}$ cell arriving to the SC. We use subscripts to identify the SC operating variables associated with cell j. That is, t_j is the arrival time of cell j, and TRT_j is its earliest scheduled departure time. We take the value of LRT_j to be the time of the last cell departure prior to the arrival of the $j+1^{st}$ cell (recall that since arriving cells might be either discarded or queued, this last departure need not be the j^{th} arrival).

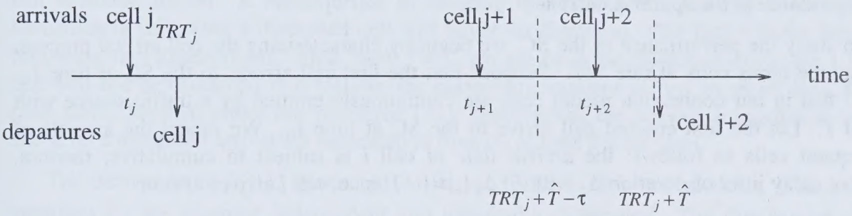


Figure 2 - A time line illustrating Spacer-Controller actions taken on a particular arrival sample path. Cell j arrives just prior to its earliest scheduled departure (TRT_j) and is delayed, then emitted. Cell $j+1$ arrives within $T-\tau$ time units of the last departure, and is discarded. Cell $j+2$ arrives after $T-\tau$ time units since the last departure, and is delayed, then emitted.

Note in Figure 1 that the value of the Theoretical Re-emission Time (TRT) of the current cell is initially set to the sum of *LRT* and $\hat{T}$. But if the cell arrives "late," the value of *TRT* is reset to the cell's arrival time. To distinguish between the values of the variable *TRT* at these different points in the algorithm, we label the initial value of *TRT* as TRT'. Hence, for cell j we have

$$TRT_j' = LRT_j + \hat{T}. \tag{3.1}$$

If cell j is not discarded, the value of TRT_j will be set to

$$TRT_j = \max\,[TRT_j' \,,\, t_j]\,. \tag{3.2}$$

The operation of the SC can then be described by the following equations:

$$LRT_j = \begin{cases} \max\,[TRT_j' \,,\, t_j] = TRT_j & \textit{if cell j is not discarded} \\ LRT_{j-1} & \textit{if cell j is discarded} \end{cases} \tag{3.3}$$

$$TRT_{j+1}' = \begin{cases} \hat{T} + \max\,[TRT_j' \,,\, t_j] = \hat{T} + TRT_j & \textit{if cell j is not discarded} \\ \hat{T} + LRT_{j-1} & \textit{if cell j is discarded.} \end{cases} \tag{3.4}$$

Recall that the first arriving cell departs immediately (i.e., suffers no delay). Hence, $TRT_0' = t_0$. Using this fact and (3.4) we can show that if none of the first j cells is discarded, than TRT_j' is given by

$$TRT'_j = \max[t_0 + j\hat{T},\ t_1 + (j-1)\hat{T},\ t_2 + (j-2)\hat{T},\ \cdots,\ t_{j-1} + \hat{T}]. \quad (3.5)$$

Eq. 3.5 is simply the nonrecursive form of (3.4) when no previous cells are discarded. Define a new collection of random variables as

$$\zeta_j \triangleq TAT'_j - t_j \qquad j=1,2,\cdots, \qquad (3.6)$$

with corresponding distributions $f_{\zeta_j}[k] \triangleq P[\zeta_j = k]$. The variable ζ measures the difference between a cell's expected and actual arrival times; the distribution f_ζ will determine the connection's cell delay and discard probabilities, and hence the connection service quality.

4. Performance of the Spacer-Controller

To study the performance of the SC, we begin by characterizing the cell arrival process. Our source emits cells at rate $1/T$. Suppose that the first cell arrives to the SC at time t_0. Recall that in our connection model cells are continuously emitted by a traffic source with period T. Let the first emitted cell arrive to the SC at time t_0. We model the arrivals of subsequent cells as follows: the *arrival time* of cell i is subject to cumulative, random, additive delay jitter of duration Δ_i with $E[\Delta_i] \equiv 0$. Hence, cell i arrives at time

$$t_i = t_0 + iT + \Delta_i\ , \qquad i=1,2,\cdots . \qquad (4.1)$$

Inserting (4.1) into (3.5) and simplifying yields the initial TAT of the $j+1^{st}$ arrival as

$$TAT'_j = t_0 + j\hat{T} + \max[(T-\hat{T})+\Delta_1,\ 2(T-\hat{T})+\Delta_2,\ \cdots,\ (j-1)(T-\hat{T})+\Delta_{j-1}]^+ \quad (4.2)$$

where $[x]^+ \triangleq max[0,x]$. The interpretation of (4.2) is straightforward. The expected arrival time of cell j is j periods of duration $\hat{T}$ after the first arrival, plus an additional time equal to the longest CDV experienced by any previous cell. Substituting (4.1) and (4.2) into (3.6) yields

$$\zeta_j = j(\hat{T}-T) - \Delta_j + \max[(T-\hat{T}) + \Delta_1,\ \cdots,\ (j-1)(T-\hat{T}) + \Delta_{j-1}]^+ . \quad (4.3)$$

In theory we can use (4.3) to determine the service quality (e.g., cell discard rate) for the connection. However, to do so is difficult, since this requires a complete specification of the entire delay jitter process. Hence, we proceed as follows - we show that for a rather special case we can use (4.3) to obtain an exact analytical result for the cell discard rate. In the followup section, we will turn to simulation to study more general arrival processes.

The special case we consider is when the source negotiates a peak cell rate equal to the actual cell emission rate ($T=\hat{T}$), and each cell in the stream is subject to an independent delay variation. We then have that (4.3) reduces to

$$\zeta_j = [\max(\Delta_1, \Delta_2, \cdots, \Delta_{j-1})]^+ - \Delta_j . \qquad (4.4)$$

We model the induced delay jitter on each cell arrival as independent, identically distributed (i.i.d.), zero-mean random variables (i.e., $\Delta_i \equiv \Delta$ with $E[\Delta] = 0$). We can then define a cell delay variation distribution

$$f_\Delta[k] \triangleq P[\Delta=k], \qquad k = \cdots, -2, -1, 0, 1, 2, \cdots\ slots, \qquad (4.5)$$

and a cumulative CDV distribution $F_\Delta[k] \triangleq P[\Delta \le k]$. The validity of this independence assumption will clearly depend on the dynamics of other present traffic, which we assume unknown. For example, if the CDV is introduced on a private access network comprising a large number of multiplexors receiving inputs from a large number of independent, relatively low-rate traffic sources, the assumption may represent a reasonable approximation. A more likely situation, however, is having multiplexor buffers introducing correlations in the delays of consecutive cells, possibly making the independence assumption inappropriate.

Certain restrictions must be imposed on the CDV distribution. It is reasonable to assume that the CDV magnitude is bounded by some delay η (i.e., $f_\Delta[k] = 0, |k| > \eta$). We would further like our model to represent anticipated ATM network connections where **1)** at most 1 cell arrives to the policer in each time slot, and **2)** cells arrive to the SC in the order in which they are generated (i.e., sequenced delivery). These restrictions on cell arrivals are inherent characteristics of ATM networks. It is apparent from (4.1) and (4.5) that we must therefore restrict $\eta < \lfloor T/2 \rfloor$. That is, the delay induced on any cell can not exceed half a cell emission period. A consequence of limiting η is that a cell arriving to the policer immediately following a discarded cell will never be discarded. This fact follows from the following two observations. First, with $\hat{T} = T$, we see from the flowchart of Figure 1 that a cell will not be discarded if it follows the previous compliant arrival by more than $T - \tau$ time slots. Second, under our restricted CDV distribution, the arrivals immediately before and after a discarded arrival are separated in time by at least T time slots.

The derivation of $f_{\zeta_j}[k]$ given by (4.4) is as follows. The arrival delay jitter random variables Δ_i are assumed independent and identically distributed. The distribution of the greater of n i.i.d. random variables [10] with cumulative distribution $F[k]$ is given by $(F[k])^n - (F[k-1])^n$. It follows that

$$P[\max(\Delta_1, \Delta_2, \cdots, \Delta_{j-1}) = k] = (F_\Delta[k])^{j-1} - (F_\Delta[k-1])^{j-1}. \tag{4.6}$$

A simple transformation yields

$$P[\,[\max(\Delta_1, \Delta_2, \cdots, \Delta_{j-1})]^+ = k\,] = \tag{4.7}$$

$$\delta[k] \sum_{i \le 0} [(F_\Delta[i])^{j-1} - (F_\Delta[i-1])^{j-1}] + [(F_\Delta[k])^{j-1} - (F_\Delta[k-1])^{j-1}]u[k-1]$$

where $\delta[k] \triangleq \begin{cases} 1 \text{ if } k=0 \\ 0 \text{ if } k \ne 0 \end{cases}$, and $u[k] \triangleq \begin{cases} 1 \text{ if } k=0,1,2,\cdots \\ 0 \text{ if } k=-1,-2,\cdots \end{cases}$.

Eq. 4.4 is the difference between two independent random variables, so the desired distribution is simply the convolution of (4.7) and the distribution of $-\Delta$:

$$f_{\zeta_j}[k] = f_{-\Delta}[k] * \left\{ \delta[k] \sum_{i \le 0} [(F_\Delta[i])^{j-1} - (F_\Delta[i-1])^{j-1}] \right. \tag{4.8}$$

$$\left. + [(F_\Delta[k])^{j-1} - (F_\Delta[k-1])^{j-1}]u[k-1] \right\}.$$

Solving for this distribution enables us to find various quality of service measures for our connection. We first find the probability that cell j is discarded, conditioned on the event that none of the cells prior to cell j are discarded. This event occurs when a cell arrives more than τ time slots earlier than its expected arrival time (i.e., $TAT_j' - t_j > \tau$). Hence we have

$$P[\zeta_j > \tau] = \sum_{k > \tau} f_{\zeta_j}[k]. \tag{4.9}$$

We can then use (4.9) to find the unconditional probability that a cell is discarded. Note that if cell j is the first discarded, the number of cells between discards will be j. Further, at most 1 cell can be discarded; the subsequent arrival will not be discarded. Let ϕ_j be the event that cell j is the first cell discarded, given that a cell is discarded. From (4.9) we can write this conditional probability as

$$P[\phi_j] = \frac{\sum_{l>\tau} f_{\zeta_j}[l]}{\sum_{i=1}^{\infty}\sum_{k>\tau} f_{\zeta_i}[k]}. \qquad (4.10)$$

We next define the duration of a departure busy period (from the policer) as the number of cell departures between discards. It follows that the expected number of cells in a departure busy period is $\sum_{j=1}^{\infty} j\,P[\phi_j]$. If we assume that the cell departure process is ergodic, we can write the average probability that a cell from our periodic source is discarded as

$$\frac{E[discards\ between\ busy\ periods]}{E[discards\ between\ busy\ periods] + E[cells\ in\ a\ busy\ period]} = \frac{1}{1+\sum_{j=1}^{\infty} jP[\phi_j]}. \qquad (4.11)$$

We can now also write other desired performance measures, such as cell delay probabilities. The probability that cell j is delayed is given by

$$P[0 < \zeta_j \le \tau] = \sum_{k=1}^{\tau} f_{\zeta_j}[k]. \qquad (4.12)$$

The conditional delay distribution of cell j is given by

$$P[\zeta_j = k \mid 0 < \zeta_j \le \tau] = \frac{f_{\zeta_j}[k]}{\sum_{i=1}^{\tau} f_{\zeta_j}[i]}, \qquad k=1,2,\cdots,\tau. \qquad (4.13)$$

Eqs. 4.11-13 provide us with a direct relationship between the service quality obtained by our traffic source and the distribution of cell delay variations introduced in the cell stream. However, they do not tell us directly how we might adjust the cell emission rate to maintain a desired service quality - recall that (4.11-13) were derived assuming $T = \hat{T}$. Fortunately, some very simple insights can be obtained by examination of (4.11). If the negotiated period $\hat{T}$ is chosen sufficiently large that $T - \hat{T} > \eta$, then the process ζ_j is nonpositive, and no cell will be discarded. However, this selection of negotiated rate $\hat{T}$ would likely be prohibitively expensive since it can be as much as 50% higher than the source emission rate.

5. Simulation Results

To extend the analysis of the previous section, we simulated the operation of a Spacer-Controller receiving a cell stream from a jittered periodic traffic source. The cell delay variation in each cell interarrival period was modeled as independent, uniform random variables with distribution width n.

In the previous section we showed that a source can negotiate a peak rate larger than the cell generation rate to achieve a lower cell discard probability. We now consider an equivalent formulation, which is to determine the cell emission period required to achieve long average departure busy periods for a fixed negotiated rate. We simulated a source which negotiated a cell arrival period of $\hat{T} = 100$ slots, and CDV tolerances of 9, 19, 29 and 39 slots. Under this traffic descriptor, Figure 3 depicts the actual cell emission period, $\hat{T}$, required to achieve relatively long busy period durations.

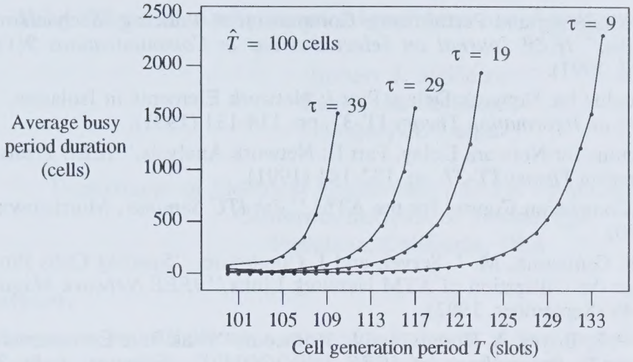


Figure 3 - Average departure busy period duration as the negotiated cell emission rate and CDV tolerance are increased.

6. Future Research Problems

There are a variety of interesting research problems that remain in peak rate enforcement and traffic shaping. The most significant challenge is to understand the role of peak rate policing and traffic shaping in the context of an overall framework of traffic controls and quality of service requirements [11, 12]. A first step must be to characterize the environment in which policing and shaping are in fact required, rather than merely an optional traffic regulation tool. A second step is to determine whether the Spacer-Controller is the appropriate regulator to operate in those regimes.

One desirable study would be to determine the Spacer-Controller delay and loss performance when consecutive cell interarrival periods are not independent. One possible analytical approach would be to consider the Markov model of jittered traffic studied by Roberts [13]. Alternatively, one might consider simulating the SC with a jittered input stream derived from an actual application (e.g., CBR traffic passed through multiple heavily loaded multiplexors).

A second avenue of investigation is to find simpler implementations of traffic shapers. Perhaps more importantly, a design of an access regulator supporting multiple enforcement and traffic regulation functions is sought. A simple implementation is required, since it is likely that as many as a few thousand regulators will be required at each regulation point (e.g., at the UPC).

It is also necessary to construct simple engineering rules to properly dimension policers to achieve a desired QoS. Finally, the Spacer-Controller is an "open loop" implementation of a regulator. An interesting avenue of investigation would be the design of "closed loop" regulators. This type of regulator would provide feedback to sources informing them of peak rate violations. The feedback could be used by the source to dynamically renegotiate its traffic descriptors and parameters.

Acknowledgement: Special thanks to A. Berger, R. Cruz, S. Low, N. Maxemchuk for helpful discussions.

1. E. P. Rathgeb, ''Modeling and Performance Comparison of Policing Mechanisms for ATM Networks,'' *IEEE Journal on Selected Areas in Communications* **9**(1), pp. 325-334 (April, 1991).
2. R. Cruz, ''A Calculus for Network Delay, Part I: Network Elements in Isolation,'' *IEEE Transactions on Information Theory* **IT-37**, pp. 114-131 (1991).
3. R. Cruz, ''A Calculus for Network Delay, Part II: Network Analysis,'' *IEEE Transactions on Information Theory* **IT-37**, pp. 132-141 (1991).
4. P. E. Boyer, ''A Congestion Control for the ATM,'' *7th ITC Seminar*, Morristown, NJ, (October, 1990).
5. P. E. Boyer, F. M. Guillemin, M. J. Servel, and J. Coudreuse, ''Spacing Cells Protects and Enhances the Utilization of ATM Network Links,'' *IEEE Network Magazine* **6**(5), pp. 38-49 (September, 1992).
6. F. M. Guillemin, P. E. Boyer, A. Dupuis, and L. Romoeuf, ''Peak Rate Enforcement in ATM Networks,'' *Proceedings of IEEE INFOCOM'92*, Florence, Italy **2**, pp. 753-758 (May, 1992).
7. A. I. Elwalid and D. Mitra, ''Analysis and Design of Rate-Based Congestion Control of High Speed Networks, I: Stochastic Fluid Models, Access Regulation ,'' *Queueing Systems* **9**, pp. 29-64 (1991).
8. CCITT Study Group XVIII Draft Recommendation I.371, *Traffic Control and Resource Management in B-ISDN*, June, 1992.
9. CCITT Study Group XVIII Temporary Document 31, *Rapporteur on Traffic Control and Resource Management*, December, 1991.
10. C. W. Helstrom, *Probability and Stochastic Processes for Engineers*, Macmillan Publishing Co. , New York (1984).
11. P. Billingsley, *Probability and Measure*, John Wiley & Sons, New York (1986).
12. A. E. Eckberg, B. T. Doshi, and R. Zoccolillo, ''Controlling Congestion B-ISDN/ATM: Issues and Strategies ,'' *IEEE Communications Magazine* **29**, pp. 64-70 (September, 1991).
13. A. E. Eckberg, ''B-ISDN/ATM Traffic and Congestion Control,'' *IEEE Network Magazine* **6**(5), pp. 28-37 (September, 1992).
14. J. W. Roberts and F. M. Guillemin, ''Jitter and Bandwidth Enforcement,'' *IEEE GLOBECOM'91 Conference Record*, Phoenix, Arizona (December, 1991).

Why Channel Assignment Algorithms are Asymptotically Smart*

Robert J. McEliece
Eugene R. Rodemich
Sanjeev K. Deora

Department of Electrical Engineering and Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California, USA

Abstract.

In this paper, we will discuss a surprising property, shared by many channel assignment algorithms for cellular communication networks, which we call "asymptotic smartness." If M denotes the maximum possible carried traffic for a given cellular network (obtained, perhaps, by artificially placing as many customers as possible in the network), we say that a given channel assignment algorithm is asymptotically smart if, under a natural model for the offered traffic, when the intensity of the offered traffic is very large, the carried traffic for the algorithm approaches M. We will argue that asymptotic smartness is an unlikely and surprising phenomenon, but prove, nevertheless, that all algorithms of a very general type (the "maximal rank reflexive Markov" algorithms) are asymptotically smart. Finally, we will briefly discuss the relationship of our results to the "simulated annealing" technique of combinatorial optimization.

1. What is Asymptotic Smartness?

We begin with a simple example, the one-dimensional cellular network illustrated in Figure 1. We assume there are a very large number N of cells, arranged linearly, and that these cells share a common set of n channels. We further assume that the channel "re-use constraint" is that no channel may be used simultaneously in two adjacent cells. It follows that each channel can be used at most $\lceil N/2 \rceil$ times, so that the total number of calls in progress at any one time is at most $n\lceil N/2 \rceil$. This bound can actually be achieved. For example, if in each odd numbered cell a call is in progress in each of the n channels, and in each even numbered cell there are no calls in progress at all, the reuse constraint will be satisfied, and $n\lceil N/2 \rceil$ calls will be in progress. Thus the maximum possible (normalized) carried traffic for the network of Figure 1, as measured in Erlangs per channel per cell, is $\lceil N/2 \rceil / N \approx 1/2$, although the assignment of channels that achieves this maximum is rather artificial, and unlikely to occur naturally.

Next, let us assume that in each cell the offered traffic is described by a simple Poisson birth-death process, for which the rate of call request *arrivals* is λ per second, and the rate of call *departures* is μ per second, so that, in each cell, the offered traffic intensity is $\rho = \lambda/\mu$ Erlangs. We also assume that the traffic is identical for each cell, and independent from cell to cell. Let A be an arbitrary channel assignment algorithm. By definition, Algorithm A *must* block a call request, if accepting it would violate the

---

* Rodemich's contribution to this paper, and a portion of McEliece's contribution, was done at the Jet Propulsion Laboratory, California Institute of Technology, under contract to the National Aeronautics and Space Administration. Deora's contribution, and a portion of McEliece's contribution, was sponsored by a grant from GTE Laboratories. A portion of McEliece's contribution was also supported by AFOSR grant no. 91–0037 and a grant from Pacific Bell.

reuse constraints, but *may* reject it even if it does not. The *carried traffic* function $T_A(\rho)$ for algorithm A is defined to be the expected number of calls in progress in the network, under the operation of Algorithm A, as a function of the offered traffic intensity ρ.

As we saw above, the carried traffic can under no circumstances exceed $1/2$ Erlangs per channel per cell, and of course the carried traffic cannot exceed the offered traffic, so that $T_A(\rho) \leq \min(\rho, 1/2)$ for any channel assignment algorithm. In fact, using the results of [5] or [7], it is possible to show that there exist channel assignment algorithms with the property that as the number of channels n becomes arbitrarily large, the corresponding carried traffic functions approache the upper bound $\min(\rho, 1/2)$ arbitrarily closely. For this reason, $\min(\rho, 1/2)$ is called the *asymptotic carried traffic function* for the cellular network (see Figure 2).

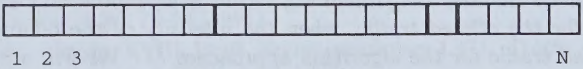


Figure 1. A one-dimensional cellular network with N cells. There are n channels, and no channel may be used simultaneously in two adjacent channels.

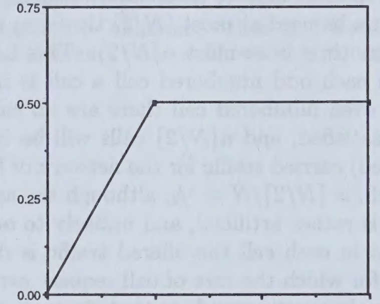


Figure 2. The asymptotic carried traffic function for the network of Figure 1 ($N = \infty$).

In order to isolate one illustrative example of "asymptotic smartness," let us now assume that $n = 1$, and $N = \infty$ in the network of Figure 1, i.e., there is only one channel available, but there are an infinite number of cells. We consider a specific channel assignment algorithm for this network, which we shall call *Algorithm G* (for "greedy"). When Algorithm G is operating, the channel is said to be *available* in a given cell if it is not in use in either of the adjacent cells. When a call request arrives at a particular cell, it is assigned to the channel if the channel is available in that cell, and otherwise it is blocked. (Algorithm G is a special case of the "DCAA" algorithm considered in [7], and the "timid" distributed dynamic channel assignment algorithm considered in [2].)

If we denote by $T_G(\rho)$ the carried traffic function for Algorithm G, it is possible to show, using techniques from [4] or [2], Appendix B, that $T_G(\rho)$ is given by the formula

$$T_G(\rho) = \frac{1}{2}\left(1 - \frac{1}{\sqrt{4\rho + 1}}\right). \tag{1}$$

This function is plotted in Figure 3, where for reference we also show the asymptotic carried traffic function for the same network from Figure 2. Note that as $\rho \to \infty$, $T_G(\rho) \to 1/2$. In other words, as the offered traffic intensity becomes very large, Algorithm G somehow manages to arrange the accepted calls into a pattern which achieves the largest possible carried traffic for the network. We therefore say that Algorithm G is "asymptotically smart".

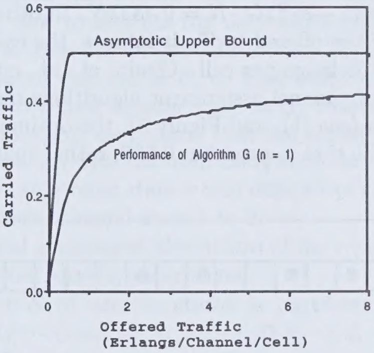


Figure 3. The carried traffic function for Algorithm G, compared to the asymptotic carried traffic function from Figure 2.

More generally, we say that a channel assignment algorithm is asymptotically smart if it has the property that as the offered traffic intensity becomes very large, the algorithm eventually coaxes the network into a state which is globally maximal. In the next section we will argue that asymptotic smartness is an unlikely phenomenon, since it is quite easy to construct networks for which the carried traffic function $f(s)$ has many local, but few global, maxima. But then in Section 3, we will describe a very general class of channel assignment algorithms, called the *reflexive Markov* algorithms, and in Section 4 we will give a proof that all such channel assignment algorithms are asymptotically smart. Finally, we will make some brief remarks about the relationship of our results and the "simulated annealing" technique of combinatorial optimization.

2. Why Is Asymptotic Smartness Surprising?

In this section we will argue that asymptotic smartness is patently unlikely, again by considering the one-dimensional, one channel network of Figure 1. To simplify the example still further, we consider the case of *infinite* offered traffic, which means that the mean time between call request arrivals is infinitesmal compared to the mean call duration. We model this by assuming that once a call arrives, it never leaves the system.

Under this model, the behavior of Algorithm G can be described as follows. We indicate the presence of a call in progress in a cell by placing a marker in that cell. There is a discrete clock, and at each tick of the clock, a cell is chosen at random. If that cell is unmarked, and if the two adjacent cells are also unmarked, we mark the cell. Otherwise, we do nothing, and move on to the next cell. Eventually the system will reach, and remain forever in, a *saturated state*, like that shown in Figure 4. In a saturated state, no two consecutive cells can be marked, and no three consecutive cells can be unmarked. Among all saturated states, the maximum possible number of occupied cells in a saturated state is of course $\lceil N/2 \rceil$, which occurs when every other cell is occupied. But when call requests arrive at random, one cannot expect to be so lucky, and a sub-maximal saturated state like that shown in Figure 4 is more likely. In fact, Cimini et. al. [2] have shown that if N is large, the *average* saturated occupancy of an N cell system aproaches $(1 - e^{-2})/2 \cdot N = 0.4323N$. In other words, for the one-channel system of Figure 1, if the offered traffic is infinite, the carried traffic saturates at an average value of 0.4323 Erlangs per cell. Cimini et. al. cite this calculation as evidence that certain classes of channel assignment algorithms may inevitably be sub-optimal, since as we have seen (see (1), and Figure 3), the optimal channel assignment algorithms have a carried traffic that approaches 0.500, rather than 0.4323, Erlangs per cell.

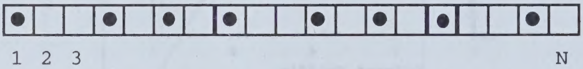


Figure 4. A locally maximal one-dimensional, one-channel cellular system.

Stated another way, for this example we have the paradoxical result that $T_G(\rho)$ is a monotonically increasing function of r (see eq. (1)), yet

$$T_G(\infty) < \lim_{\rho \to \infty} T_G(\rho). \tag{2}$$

Stated still another way, even though the network of Figure 1 contains many states like that in Figure 4 that are locally but not globally maximal, nevertheless Algorithm G somehow asymptotically manages to avoid such states, and, on average, occupies a near-maximal configuration. This is why asymptotic smartness is surprising.

3. Reflexive Markov Channel Assignment Algorithms.

In this section, we will describe a very general class of channnel assignment algorithms, which we call the *maximal rank reflexive Markov* algorithms, which all turn out to be asymptotically smart.

We consider only cellular networks with a finite number, N, of cells, and a finite number, n, of channels. We assume the standard model for the offered traffic as described in Section 1, i.e., an independent Poisson birth-death process with mean value ρ in each cell. The number ρ is called the *intensity of the offered traffic.*

We define the *atomic state* of the network as the $n \times N$ matrix S with $S_{i,j} = 1$ if channel i is active in cell j, and $S_{i,j} = 0$ if not. The *rank* of the state S, denoted by rank(S), is defined to be the total number of 1's in S, which corresponds to the number of calls in progress when the network is in state S. We also define the *type* of S to be the N-tuple $(n_1, \ldots, n_N)$, where $n_j = \sum_{i=1}^{n} S_{ij}$, i.e., the number of calls in progress in cell j when the network is in state S. There are 2^{nN} possible atomic states, but for a given network, only a subset of these will be permissible, because of the channel reuse constraints. In fact, we shall *define* a "reuse constraint" to be subset of the atomic states which is closed under the operation of changing a 1 in the state matrix to a 0, i.e., removing a caller from the system.

If $\mathcal{S}$ is an arbitrary reuse constraint for an N-cell, n-channel, cellular network of the type just defined, we define the rank of $\mathcal{S}$ to be the maximum possible value of the rank for $\mathcal{S}$, i.e., $\text{rank}(\mathcal{S}) = \max_{S \in \mathcal{S}} \text{rank}(S)$.

A channel assignment algorithm A for $\mathcal{S}$ is a rule for deciding, whenever a call request arrives in a given cell, whether to accept or block the call; and if the decision is to accept the call, how to allocate the networks's resources to accomodate the new call without having to drop any of the calls already in progress. The network's atomic state must always be an element of $\mathcal{S}$, so that when a call is accepted in cell j, the network must move to a new atomic state in $\mathcal{S}$, which has exactly one more call in cell j. If the call is blocked, we assume that the atomic state of the network remains unchanged. When a caller, who has been using channel i in cell j, hangs up, we assume the network moves to an atomic state which differs from the previous state only in that the (i, j) entry has been changed from 1 to 0.

Among all channel assignment algorithms of the type just describe, we now identify a subclass, the *reflexive Markov* algorithms.

Suppose the set $\mathcal{S}$ of atomic states is partitioned into a number of subsets $\sigma_1, \ldots, \sigma_m$, i.e., $\sigma_1 \cup \cdots \cup \sigma_m = \mathcal{S}$, and $\sigma_i \cap \sigma_j = \emptyset$, if $i \neq j$. We call the σ_i the *molecular* states with respect to the partition. Assume further that this partition respects the "type" function, i.e., all atomic states in a given molecular state have the same type. For example, the molecular states could be the same as the atomic states; or they could be the subsets of atomic states of constant type. However, since there is only one atomic state of type $(0, 0, \ldots, 0)$, viz. the all-zero state, this rank-0 atomic state, which we call the *ground state*, will always be one of the molecular states. In what follows, we will refer to the molecular states simply as the *states.*

When a given channel assignment algorithm is in operation, as time passes, the network will move among the states in a random manner. Call departures, which are beyond the control of the algorithm, cause the rank of the state to decrease by one, but call acceptances, which are under the control of the algorithm, cause the rank to increase by one. The *rank of the algorithm* is defined to be the maximum rank of any state that can be reached from the ground state under the operation of the algorithm. The algorithm is said to be of *maximal rank* if its rank is the same as rank($\mathcal{S}$). (Plainly no algorithm can be asymptotically smart unless it is of maximal rank.)

We assume that when the system is in state σ, and a call request arrives in a cell j, that the next state is determined, possibly probabilistically, in a manner that depends only on the pair (σ, j), and not, e.g., on the intensity of the offered traffic or the history of the system prior to its arrival in state σ. We denote the probability of a transition from state σ to state σ', given the occurrence of a call request in cell j, by $p(\sigma'|\sigma, j)$. This is the *Markov assumption.* Also, we assume that whenever there is a positive probability of moving from state σ to state σ', there also a positive probability of moving from state σ' to σ. This is the *reflexive assumption.* An algorithm which satisfies both the Markov assumption and the reversibility assumption is called a *reflexive Markov algorithm.*

When a reflexive Markov algorithm is operating, the movement of the network among the molecular states is a continuous-time Markov chain, say $X_\rho(t)$. The *upward* transition rates, i.e., those that increase the rank by one, are all proportional to ρ, reflecting the constant state transition probabilities, and the variable frequency of call arrivals. Indeed, if the transition from state σ to σ' corresponds to the acceptance of a call request in cell j, then the rate of this transition is just $\rho \cdot p(\sigma'|\sigma, j)$. On the other hand the *downward* transition rates, i.e., those that decrease the rank by one, are constants independent of ρ, reflecting the fact that the average call duration is independent of ρ. We can thus define the *asymptotic rank* of the algorithm as

$$\text{asymp.rank} = \lim_{\rho \to \infty} E(\text{rank}(X(t)). \tag{3}$$

To say that an algorithm is asymptotically smart is then to say that its asymp.rank is the same as the rank of the network. Here is the main result of the paper.

Theorem 1. *If A is a reflexive Markov algorithm its* asymp.rank *is equal to its rank. In particular, every maximal-rank reflexive Markov algorithm is asymptotically smart.*

For example, Algorithm G, for the cellular system of Figure 1, is an MRRM algorithm with respect to the atomic states. Similarly, all of the distributed dynamic channel assignment algoritms of [1] are MRRM with respect to the atomic states. The "maximum packing" algorithm of [4], and the more general "Ω" algorithms defined in [6] are MRRM, with respect to the states of constant type. On the other hand, the channel assignment algorithms used in [5] and [7] to achieve the asymptotic carried traffic function (as $n \to \infty$) are not Markov, since their operation depends on the value of ρ. As a final example, consider the system of Figure 1 with $n = 2$, and $N = 3$, in which a call is accepted in cell 2 whenever there is an available channel in that cell, but a call is accepted in cells 1 or 3 only when there is an available channel in that cell, and after accepting the call, and there is also a channel available in cell 2. (If more than one channel is available, one is chosen at random.) This algorithm has rank 3, and is Markov, but it is not reflexive, since if

$$S_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}, \qquad S_2 = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix},$$

it is possible for the algorithm to move from S_2 to S_1 (when a call departs from cell 3), but it is not possible for the algorithm to move from S_1 to S_2. The asymptotic rank of this algorithm can be shown to be 2, so that the "reflexive" assumption is necessary in Theorem 1.

In the next section, we will prove a general theorem about "scaled" reflexive Markov chains, which will imply Theorem 1.

4. Why Maximal Rank Reflexive Markov Algorithms Are Asymptotically Smart.

We begin with a connected graph $G = (V, E)$, with p vertices, which is "graded" in the sense that attached to each vertex v is a nonnegative integer, called the *rank* of v. If v and v' are vertices connected by an edge, then $|\operatorname{rank}(v) - \operatorname{rank}(v')| = 1$. In Figure 5, we see a graded graph with 6 vertices.

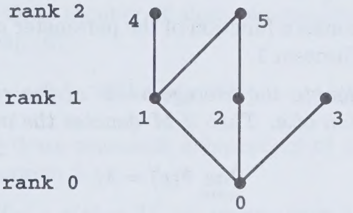


Figure 5. A connected, graded graph, that serves as the state diagram for a continuous-time Markov chain.

We use G as the state diagram for a continuous-time Markov chain, with direct state transitions only allowed between adjacent states, and the "rank" of a vertex serving as a generalization of the notion of carried traffic. Furthermore, we assume that the chain is reflexve, i.e, if v and v' are connected by an edge, that the transition rates from v to v' and from v' to v, are both positive. (This insures that the associated Markov chain is irreducible, since G is connected.) We can display the transition rates for the Markov chain in a $p \times p$ matrix A, with the transition rate from vertex v to vertex v' appearing as the (v, v') entry of A. It is convenient to define the diagonal entries of A so that the row sums of A all equal to zero. If this is done, and if we denote by $q(v)$ the steady-state probability of the process being in vertex v, then the vector $(q(v)_{v \in V})$ is a basis for the (one-dimensional) left nullspace of A. (See [3], Chapter 6, or [9], Chapter 3). For example, for the graph of Figure 6, we might have the following matrix of transition rates.

$$(4) \qquad A = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{array}{c} \begin{array}{cccccc} 0 & 1 & 2 & 3 & 4 & 5 \end{array} \\ \begin{pmatrix} -5 & 1 & 1 & 3 & 0 & 0 \\ 3 & -5 & 0 & 0 & 1 & 1 \\ 2 & 0 & -3 & 0 & 0 & 1 \\ 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 3 & 0 & 0 & -3 & 0 \\ 0 & 2 & 3 & 0 & 0 & -5 \end{pmatrix} \end{array}$$

The vector $(18, 6, 9, 54, 2, 3)$ is a basis for the left nullspace of A, (the command "`NullSpace[Transpose[A]]`" in *Mathematica* produces this vector), and so the steady-state probabilities for the Markov chain associated with A are $q(0) = {}^{18}\!/_{92}$, $q(1) = {}^{6}\!/_{92}$,

$q(2) = 9/92$, $q(3) = 54/92$, $q(4) = 2/92$, and $q(5) = 3/92$, so that the average rank is $79/92 = 0.8578$. Let us now "stimulate" transitions in the upward direction, by multiplying each upward transition by a scale factor of ρ, and call the new transition matrix $A(\rho)$. Thus for example, the matrix A in eq. (4) becomes

$$A(\rho) = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{array}{c} \begin{array}{cccccc} 0 & 1 & 2 & 3 & 4 & 5 \end{array} \\ \begin{pmatrix} -5\rho & \rho & \rho & 3\rho & 0 & 0 \\ 3 & -3-2\rho & 0 & 0 & \rho & \rho \\ 2 & 0 & -2-\rho & 0 & 0 & \rho \\ 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 3 & 0 & 0 & -3 & 0 \\ 0 & 2 & 3 & 0 & 0 & -5 \end{pmatrix} \end{array} \tag{5}$$

The average rank now becomes a function of the parameter ρ, and we have the following theorem, which implies Theorem 1.

Theorem 2. *Let $\overline{r}(\rho)$ denote the average value of the rank of a graded, reflexive, Markov chain, as a function of ρ. Then if M denotes the maximum rank of the graph G, then*

$$\lim_{\rho\to\infty} \overline{r}(\rho) = M$$

The idea of the proof of Theorem 2 is to show that if $q(v,\rho)$ denotes the steady-state probability of vertex v, as a function of ρ, then as $\rho \to \infty$, $q(v,\rho) \to 0$ *unless* v has maximum rank. For example, consider the parameterized chain described by eq. (5). Then we have, by direct calculation, that a basis for the left nullspace of $A(\rho)$ is $(18, 6\rho, 9\rho, 54\rho, 2\rho^2, 3\rho^2)$, and so, as $\rho \to \infty$,

$$\begin{aligned} q(0,\rho) &= \frac{18}{5}\rho^{-2} + O(\rho^{-3}) \\ q(1,\rho) &= \frac{6}{5}\rho^{-1} + O(\rho^{-2}) \\ q(2,\rho) &= \frac{9}{5}\rho^{-1} + O(\rho^{-2}) \\ q(3,\rho) &= \frac{54}{5}\rho^{-1} + O(\rho^{-2}) \\ q(4,\rho) &= \frac{2}{5} + O(\rho^{-1}) \\ q(5,\rho) &= \frac{3}{5} + O(\rho^{-1}), \end{aligned} \tag{6}$$

which shows how the "mass" of the process concentrates on the vertices of maximum rank, i.e., vertices 4 and 5, as $\rho \to \infty$. This proves Theorem 2 for this one example.

In fact, Theorem 2 is itself a special case of a still more general result about continuous-time Markov chains. Let $X(t)$ be an irreducible, continuous-time Markov chain, taking values in the finite set V, and let $\lambda(v,v')$ denote the transition rate from vertex v to vertex v'. Assume that $X(t)$ is reflexive, i.e., $\lambda(v,v') \neq 0$, if and only if $\lambda(v',v) \neq 0$. Let $f(v)$, $v \in V$, be an arbitrary real-valued function, and let $\rho \geq 1$ be a parameter. Define the *parameterized chain* $X_\rho(t)$ to be the Markov chain on V whose transition rates are given by

$$\lambda_\rho(v,v') = \rho^{\max(f(v')-f(v),0)}\lambda(v,v'). \tag{7}$$

Thus in the chain X_ρ, transitions that decrease f are not affected, but those that increase f are amplified, the amplification factor being $\rho^{\Delta f}$.

Theorem 3. *Denote by $q(v, \rho)$ the steady-state probabilities of the reflexive chain X_ρ, i.e., $q(v, \rho) = \lim_{\tau \to \infty} \Pr\{X_\rho(t) = v\}$, and let $M = \max_{v \in V} f(v)$. Then*

$$\lim_{\rho \to \infty} q(v, \rho) = 0 \qquad \textit{unless } f(v) = M.$$

In other words, as ρ increases, the equilibrium probabilities are attracted to the vertex or vertices for which $f(v)$ is a global maximum.

Proof: Let $A(\rho)$ denote the matrix of transition probabilities, as given by eq. (7) for the parameterized chain X_ρ, augmented so that the row sums are zero. Then since the chain is irreducible, the vector of equilibrium probabilities is the unique vector in the left nullspace of $A(\rho)$ with component sum one. Thus the steady-state probabilities for X_ρ can be computed by a finite number of algebraic operations on the entries of $A(\rho)$, and so we must have (cf. eq. (6))

$$q(v, \rho) = c(v)\rho^{-e(v)} + o(\rho^{-e(v)}), \qquad \text{as } \rho \to \infty \tag{8}$$

where $c(v) > 0$ and $e(v) \geq 0$ are constants independent of ρ. The following Lemma is the key to the proof of Theorem 3.

Lemma. *For all $s \in V$, $f(v) + e(v) = M$, where as above $M = \max_{v \in V} f(v)$.*

Proof: Let $m = \max_{v \in V}(f(v) + e(v))$, and let $V_0 = \{v \in V : f(v) + e(v) = m\}$. In equilibrium, there cannot be a net flow into or out of V_0, but we will show that if V_0 is not the entire set V, there will be a net flow *into* V_0 for sufficiently large ρ. This will imply that $V_0 = V$, which proves that $f(v) + e(v) = m$ for all $v \in V$.

Thus suppose $V_0 \neq V$. Then, since the underlying graph G is connected, there will be one or more edges of G with one vertex in V_0, and one outside of V_0. Let (x, y) be one such edge, with $x \in V_0$, and $y \notin V_0$. By the definition of V_0, we then have

$$f(x) + e(x) > f(y) + e(y). \tag{9}$$

We now consider the "flow" along the edge (x, y), in both directions, in equilibrium. The flow from x to y is, by eq. (7) given by

$$q(x, \rho) \cdot \rho^{\max(f(y) - f(x), 0)} \lambda(x, y), \tag{10}$$

and the flow from y to x is given by

$$q(y, \rho) \cdot \rho^{\max(f(x) - f(y), 0)} \lambda(y, x). \tag{11}$$

Combining eq. (8) with eq. (10) and eq. (11), we see (since both $\lambda(x, y)$ and $\lambda(y, x)$ are nonzero), that for large values of ρ, the flow from x to y will of the form $A\rho^{-E(x,y)} + o(\rho^{-E(x,y)})$, and the flow from y to x will be of the form $A'\rho^{-E(y,x)} + o(\rho^{-E(y,x)})$, where

$$\begin{aligned} E(x, y) &= e(x) - \max(f(y) - f(x), 0) \\ E(y, x) &= e(y) - \max(f(x) - f(y), 0). \end{aligned} \tag{12}$$

We now distinguish two cases, $f(x) \geq f(y)$, and $f(x) < f(y)$. If $f(x) \geq f(y)$, then by eq. (12), we have

$$\begin{aligned} E(x, y) &= e(x) \\ E(y, x) &= e(y) - f(x) + f(y). \end{aligned}$$

By eq. (9), this implies that for sufficiently large ρ, $E(x, y) > E(y, x)$, so that there will be a net flow into V_0 along the edge (x, y). If, on the other hand, $f(x) < f(y)$, then by eq. (12), we have

$$\begin{aligned} E(x,y) &= e(x) - f(y) + f(x) \\ E(y,x) &= e(y). \end{aligned}$$

Again by eq. (9), this implies that for sufficiently large ρ, $E(x, y) > E(y, x)$, so that there will be a net flow into V_0. Thus we see that in either case, for sufficiently large ρ, there will be a net flow into V_0, which is impossible in equilibrium. Thus we have shown that $f(v) + e(v) = m$ for all $v \in V$.

To complete the proof, we must show that $m = M$. Observe from eq. (8), that if $m > M$, then $e(v) > 0$ for all $v \in V$, and so $q(v, \rho) \to 0$, as $\rho \to \infty$, which is impossible, since the $q(v, \rho)$'s are a probability distribution. On the other hand, if $m < M$, then for all v such that $f(v) = M$, we have $e(v) < 0$, which implies $q(v, \rho) \to \infty$, which is also impossible. This completes the proof of the lemma. ▪

We can now quickly complete the proof of Theorem 3. By the Lemma, we see that if $f(v) < M$, then $e(v) = M - f(v) > 0$, which implies that $q(v, \rho) \to 0$. ▪

Theorem 3 is closely related to the basic results in simulated annealing (see [8], for example). In fact, if we define the "temperature" of the chain $X_\rho(t)$ to be $T = 1/\log(\rho)$, then an expert in SA might recognize Theorem 3 as a result in his field of specialization. But we find it instructive and amusing that channel assignment algorithms exhibit naturally "smart" behavior that combinatorial optimizers must induce artificially in simulated annealing optimization. Indeed, we take the view that the asymptotic smartness of channel assignment algorithms is a better paradigm for simulated annealing than is the usual one taken from metallurgy.

References.

1. L. J. Cimini, Jr., and G. J. Foschini, "Distributed Dynamic Algorithms for Dynamic Channel Allocation in Microcellular Ststems," Proc. 1992 Vehicular Technology Conference (VTC 1992), pp. 641–644.
2. L. J. Cimini, Jr., G. J. Foschini, and L. A. Shepp, "Single-Channel User-Capacity Calculations for Self-Organizing Cellular Systems." *IEEE Trans. Comm.*, in press.
3. J. L. Doob, *Stochastic Processes.* New York: Wiley and Sons, 1953.
4. F. P. Kelly, "Stochastic Models of Computer Communication Systems," J. Royal Stat. Soc. B, Vol. 47 no. 3 (1985), pp. 379–395.
5. R. J. McEliece and K. N. Sivarajan, "Performance Limits for FDMA Cellular Telephone Systems," Proc. 1990 Allerton Conference on Communication, Control, and Computing, pp. 869–880.
6. R. J. McEliece and K. N. Sivarajan, "Heavy Traffic Performance of a Class of Channel Assignment Algorithms," Proc. Third International Symp. Personal, Indoor, and Mobile Radio Communications (PIMRC '92) (Boston, October 1992). pp. 550–555.
7. R. J. McEliece and K. N. Sivarajan, "Performance Limits for Channelized Cellular Telephone Systems," *IEEE Trans. Inform. Theory*, in press.
8. F. Romeo and A. Sangiovanni–Vincentelli, "A Theoretical Framework for Simulated Annealing," *Algorithmica*, vol. 6(1990), pp. 302–345.
9. M. Rosenblatt, *Random Processes.* New York: Oxford University Press, 1962.

Transfer Function and State Space Approaches to Rational Matrix Factorization

Madanpal S. Verma
Dept. of Electrical Engineering,
3480 University St., McGill University,
Montreal, Quebec, H2A 2A7, Canada

Abstract

Several important feedback design problems related to stabilization robustness and performance optimization can be posed in the transfer function domain. Solutions to these problems are often expressed in terms of factorizations of rational matrices. These factorizations include, for instance, coprime, inner-outer, spectral and J-inner outer factorizations of rational matrices. State space theory for computing such factorizations is further developed. It is then applied to the J-inner outer factorization problem arising in H^∞ optimization.

1 Introduction

The transfer function approach to linear feedback system design has witnessed a renaissance during the preceding decade. Many important problems related to feedback design can be posed and solved within the transfer function framework [8]. These include such problems as stabilization [9], model reduction [2], robust stabilization [7], H^2 and H^∞ performance optimization [9, 10], etc. Solution to each of these problems can be expressed in terms of factorizations of rational matrices or involves these at intermediate steps. For instance, the problem of parametrizing all stabilizing controlling for a multi-input multi-output plant requires left and right coprime factorizations of its transfer matrix [8,9]. The problem of H^2-optimization requires solution to a spectral factorization as an important intermediate step [9]. Similarly, the problems of model reduction [2], robust stabilization [7] and H^∞ optimization [1] can be solved via J-inner outer factorization. Thus a systematic theory for computing such factorizations is needed from the viewpoint of numerical computation. This will facilitate implementation of the aforementioned design techniques.

A systematic theory for computing factorizations of rational matrices is presented in the work of Bart, Grohberg, Kaashoek and Van Dooren [4]. The theory, termed minimal factorization, computes factorizations of the form $G(s) = G_1(s)G_2(s)$ where $G(s), G_1(s)$ and $G_2(s)$ are rational matrices. When there is no pole-zero cancellation between $G_1(s)$ and $G_2(s)$, or the McMillan degree of $G(s)$ is the sum of McMillan degrees of $G_1(s)$ and $G_2(s)$, then the factorization is said to be minimal. In order to compute minimal factorizations of rational matrices, transfer functions are associated with their state space realizations. The state space realization of a cascade of two realizations is derived in terms of the two realizations and its specific form examined. It turns out that the cascade gives rise to two invariant subspaces, associated with the realization of the cascade, which form a direct sum decomposition of the realization of the cascade. Conversely, finding such a pair of invariant subspaces for the realization leads to a factorization of the transfer matrix as a cascade of two transfer matrices. The elements of the cascade and their realization are determined by the invariant subspaces.

This viewpoint to computing factorizations of rational matrices has been employed in several works earlier, e.g., [1, 2]. Here we wish to apply it to computing the J-inner-outer factorization with the objective of emphasizing its numerical properties. In the past, numerical schemes for computing J-inner-outer factorization have emphasized the use of quadratic matrix equations such as the algebraic Riccati equation [4] to which it can be shown to reduced to. However, several efficient schemes for computing solution to Riccati equations are themselves based on the use of invariant subspace methods [6]. As noted earlier, solution to the J-inner-outer factorization problem can also be expressed in terms of finding a pair of invariant subspaces. Thus, it seems that developing efficient computational schemes for characterizing such invariant subspaces and for their subsequent use in computing the factorization should be a good strategy for solving the J-inner-outer factorization problem in a numerically efficient manner. This approach is explored and developed in the present work.

In section 2, the basics of transfer function factorization theory are presented. The role of invariant subspaces in computing these factorizations is discussed in section 3. The minimal factorization theory developed in [4] is then presented in section 4. This theory is then applied to the J-inner-outer factorization problem in section 5. Concluding remarks are presented in section 6.

2 Basics of Transfer Function Factorization

Let $G(s)$ be the transfer matrix of a $p \times m$ causal, finite-dimensional, linear time invariant system. Then $G(s)$ is proper and can be realized by a linear dynamical system [5] of the form

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned} \tag{2.1}$$

where $A \in \mathbb{C}^{n \times n}, B \in \mathbb{C}^{n \times m}, C \in \mathbb{C}^{p \times n}$ and $D \in \mathbb{C}^{p \times m}$. It is well-known that $G(s)$ is related to the realization in (2.1) via

$$G(s) = C(sI - A)^{-1}B + D. \tag{2.2}$$

The realization in (2.1) is minimal if (A, B) is controllable and (A, C) is observable. For a minimal realization, the McMillan degree of $G(s)$ equals n, the dimension of $G(s)$. It is well known that for a given transfer matrix, many state space realizations exist and all minimal realizations are related by a similarity transformation sketched in the following. Let $T \in \mathbb{C}^{n \times n}$ be a nonsingular matrix and let $z = Tx$ define a new state variable. Then it is easily verified that

$$\begin{aligned} \dot{z} &= TAT^{-1}z + TBu \\ y &= CT^{-1}z + Du \end{aligned} \tag{2.3}$$

is a new realization of $G(s)$ in that an equation of the form (2.2) is satisfied with $\tilde{A} = TAT^{-1}, \tilde{B} = TB, \tilde{C} = CT^{-1}$ and $\widetilde{D} = D$. In the following, we will use the notation

$$G(s) = \begin{bmatrix} A & \vdots & B \\ \cdots & \cdots & \cdots \\ C & \vdots & D \end{bmatrix} \tag{2.4}$$

to represent the relationship between $G(s)$ and its state space realization given in (2.1).

When $G(s)$ is square, i.e., $p = m$ and D is invertible, then the relationship between u and y is (2.1) can be reversed leading to

$$\begin{aligned} \dot{x} &= (A - BD^{-1}C)x + BD^{-1}y \\ u &= -D^{-1}Cx + D^{-1}u \end{aligned} \tag{2.5}$$

It can be verified that the transfer matrix associated with (2.5) is $G^{-1}(s)$ and (2.5) is called the inverse system associated with (2.1) [5]. We use the notation

$$G^{-1}(s) = \begin{bmatrix} A - BD^{-1}C & \vdots & BD^{-1} \\ \cdots & \cdots & \cdots \\ -D^{-1}C & \vdots & D^{-1} \end{bmatrix} = \begin{bmatrix} A^x & \vdots & B^x \\ \cdots & \cdots & \cdots \\ C^x & \vdots & D^x \end{bmatrix} \tag{2.6}$$

to denote the inverse system of (2.1). Let $G_i(s)$ be $p_i \times m_i$ proper rational matrices with realizations

$$\begin{aligned} \dot{x}_i &= A_i x_i + B_i u_i, \quad i = 1, 2 \\ y_i &= C_i x_i + D_i u_i \end{aligned} \tag{2.7}$$

When $m_1 = p_2$, we can form the product $G(s) = G_1(s)G_2(s)$. A realization for $G(s)$ can be obtained from (2.7) by taking $x^T = [x_1^T : x_2^T]^T$ as the state and by setting $u_1 = y_2$. This leads to

$$\begin{aligned} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} &= \begin{bmatrix} A_1 & B_1C_2 \\ 0 & A_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} B_1D_2 \\ B_2 \end{bmatrix} u_2 \\ y_1 &= \begin{bmatrix} C_1 & D_1C_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} D_1 & D_1 \end{bmatrix} u_2 \end{aligned} \tag{2.8}$$

which is a realization of $G(s)$ of dimension $n_1 + n_2$. Note that this realization of $G(s)$ may not be minimal even though the realizations of $G_1(s)$ and $G_2(s)$ in (2.7) are minimal only in the case when the realization in (2.8) is minimal, i.e., the McMillan degree of $G(s)$ is the sum of McMillan degrees of $G_1(s)$ and $G_2(s)$, the factorization $G(s) = G_1(s)G_2(s)$ is said to be a minimal factorization [4].

If both $G_1(s)$ and $G_2(s)$ are square with invertible D_1 and D_2, then both admit inverse systems $G_1^{-1}(s)$ and $G_2^{-1}(s)$. Consequently, $G^{-1}(s) = G_2^{-1}(s)G_1^{-1}(s)$ with associated realization

$$\begin{aligned} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} &= \begin{bmatrix} A_1^x & 0 \\ B_2^x C_1^x & A_2^x \end{bmatrix} \begin{bmatrix} x_i \\ x_2 \end{bmatrix} + \begin{bmatrix} B_1^x & \\ B_2^x & D_1^x \end{bmatrix} y_1 \\ u_2 &= \begin{bmatrix} D_2^x C_1^x & C_2^x \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} D_2^x D_1^x \end{bmatrix} y_1 \end{aligned} \tag{2.9}$$

The above realization for $G^{-1}(s)$ is obtained by first noting that $G_i^{-1}(s)$ admit the realization

$$\begin{aligned} \dot{x}_i &= A_i^x x_i + B_i^x y_i \\ u_i &= C_i^x x_i + D_i^x y_i \end{aligned} \tag{2.10}$$

for $i = 1, 2$. The various matrices in (2.10) are defined according to

$$\left[\begin{array}{ccc} A_i^x & \vdots & B_i^x \\ \cdots & \cdots & \cdots \\ C_i^x & \vdots & D_i^x \end{array}\right] = \left[\begin{array}{ccc} A_i - B_i D_i^{-1} C_i & \vdots & B_i D_i^{-1} \\ \cdots\cdots & \cdots\cdots & \cdots\cdots \\ -D_i^{-1} C_i & \vdots & D_i^{-1} \end{array}\right] \tag{2.11}$$

for $i = 1, 2$.

3 Invariant Subspaces

We view a matrix $A \in \mathbb{C}^{n\times n}$ as a linear transformation $A : \mathbb{C}^n \to \mathbb{C}^n$. Let M and N be subspaces of $\mathbb{C}^n$ such that $\mathbb{C}^n = M \oplus N$ and M is A-invariant [3]. Let $t_1, t_2 \cdots, t_k$ be a basis for M and $t_{k+1}, \cdots, t_k$ a basis for N. Define

$$\begin{aligned} T_1 &= [t_1 : t_2 \cdots : t_k] \\ T_2 &= [t_{k+1} : t_{k+2} \cdots : t_n] \end{aligned}$$

and

$$T = [T_1 : T_2].$$

Clearly $T \in \mathbb{C}^{n\times n}$ is nonsingular. Since M is A-invariant, there exists $\tilde{A}_{11} \in \mathbb{C}^{k\times k}$ such that

$$AT_1 = T_1 \tilde{A}_{11} \tag{3.1}$$

Thus,

$$A[T_1 : T_2] = [T_1 : T_2] \begin{bmatrix} \tilde{A}_{11} & \tilde{A}_{12} \\ 0 & \tilde{A}_{22} \end{bmatrix} \tag{3.2}$$

where $\tilde{A}_{12}$ and $\tilde{A}_{22}$ are appropriately dimensioned matrices. Hence

$$T^{-1}AT = \tilde{A} = \begin{bmatrix} \tilde{A}_{11} & \tilde{A}_{12} \\ 0 & \tilde{A}_{22} \end{bmatrix} \tag{3.3}$$

By further partitioning T_1 such that $T_{11} \in \mathbb{C}^{k\times k}$ (3.1) reduces to

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} T_{11} \\ T_{21} \end{bmatrix} = \begin{bmatrix} T_{11} \\ T_{21} \end{bmatrix} \tilde{A}_{11} \tag{3.4}$$

where the partitioning of A is consistent with that of T_1. If T_{11} is invertible, we define $X = T_{21}T_{11}^{-1}$ and write (3.4) in the form

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} I \\ X \end{bmatrix} = \begin{bmatrix} I \\ X \end{bmatrix} (T_{11}\tilde{A}_{11}T_{11}^{-1}) \tag{3.5}$$

By redefining $T_{11}\tilde{A}_{11}T_{11}^{-1}$ as $\tilde{A}_{11}$, (3.5) leads to

$$\begin{aligned} A_{11} + A_{12}X &= \tilde{A}_{11} \\ A_{21} + A_{22}X &= X\tilde{A}_{11} \end{aligned} \tag{3.6}$$

or

$$A_{21} + A_{22}X - XA_{11} - XA_{12}X = 0 \tag{3.7}$$

which is a quadratic matrix equation. If Equation (3.7) is solved for x then $\tilde{A}_{11}$ can be computed from (3.6) .

Since the subspace M is A-invariant, the restriction of the linear transformation $A : \mathbb{C}^n \to \mathbb{C}^n$ to M can be considered [3]. Note that with respect to the basis $\{t_1, t_2 \cdots, t_k\}$ of M, this restriction has a matrix representation given by $\tilde{A}_{11}$. Thus the spectrum of A restricted to M is the same as the spectrum of $\tilde{A}_{11}$.

4 Minimal Factorization

In order to compute a minimal factorization $G(s) = G_1(s)G_2(s)$ of a rational matrix, we follow the procedure developed in [4]. We assume that $G(s), G_1(s)$ and $G_2(s)$ have McMillan degrees n, k and $(n-k)$ respectively with $G(s)$ having a state space realization given by (2.1). The first step is to find a subspace M of $\mathbb{C}^n$ which is invariant under A. Based on our discussion in section 3, we assume that M is of the form

$$M = I_m \begin{bmatrix} I \\ X \end{bmatrix}$$

where X is a solution of the quadratic matrix equation

$$XA_{12}X + XA_{11} - A_{22}X - A_{21} = 0 \tag{4.1}$$

We also define

$$\tilde{A}_{11} = A_{11} + A_{12}X. \tag{4.2}$$

which gives the spectrum of A restricted to M.

The next step is to find an invariant subspace N of $A^x = A - BD^{-1}C$. We assume that N is of the form

$$N = I_m \begin{bmatrix} Y \\ I \end{bmatrix}$$

where Y is a solution of the quadratic matrix equation

$$YA_{21}^xY + YA_{22}^x - A_{11}^xY - A_{12}^x = 0 \tag{4.3}$$

and

$$\tilde{A}_{22}^x = A_{21}^xY + A_{22}^x \tag{4.4}$$

gives the spectrum of A restricted to N. We assume now that M and N form a direct sum decomposition of $\mathbb{C}^n$, i.e.

$$\mathbb{C}^n = M \oplus N.$$

Thus the $n \times n$ matrix

$$T = \begin{bmatrix} I & Y \\ X & I \end{bmatrix} \tag{4.5}$$

is nonsingular.

We use T to induce a similarity transformation on the realization (2.1) of $G(s)$. From (4.1) and (4.2), it follows that

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} I & Y \\ X & I \end{bmatrix} = \begin{bmatrix} I & Y \\ X & I \end{bmatrix} \cdot \begin{bmatrix} \tilde{A}_{11} & \tilde{A}_{12} \\ 0 & \tilde{A}_{22} \end{bmatrix} \tag{4.6}$$

Similarly, from (4.3) and (4.4), it follows that

$$\begin{bmatrix} A^x_{11} & A^x_{12} \\ A^x_{21} & A^x_{22} \end{bmatrix} \begin{bmatrix} I & Y \\ X & I \end{bmatrix} = \begin{bmatrix} I & Y \\ X & I \end{bmatrix} \cdot \begin{bmatrix} \tilde{A}^x_{11} & 0 \\ \tilde{A}^x_{21} & \tilde{A}^x_{22} \end{bmatrix} \tag{4.7}$$

where the left hand side represents a partitioning of A^x. As a consequence of (4.6) and (4.7),

$$T^{-1}AT = \begin{bmatrix} \tilde{A}_{11} & \tilde{A}_{12} \\ 0 & \tilde{A}_{22} \end{bmatrix} \tag{4.8}$$

and

$$T^{-1}A^xT = \begin{bmatrix} \tilde{A}^x_{11} & 0 \\ \tilde{A}^x_{21} & \tilde{A}^x_{22} \end{bmatrix}.$$

Since, $T^{-1}AT$ and $T^{-1}A^xT$ have the form of the product realizations given in (2.8) and (2.9), $G(s)$ admits a factorization of the form $G(s) = G_1(s)G_2(s)$. To compute this factorization, first factor

$$D = D_1 D_2$$

with square and nonsingular D_1 and D_2. Note that the choice of D_1 and D_2 is nonunique. We define next

$$T^{-1}B = \begin{bmatrix} \tilde{B}_1 \widetilde{D}_2 \\ \tilde{B}_2 \end{bmatrix}, CT = \begin{bmatrix} \tilde{C}_1 & \widetilde{D}_1 \tilde{C}_2 \end{bmatrix}$$

This leads to the realizations

$$G_i(s) = \begin{bmatrix} \tilde{A}_{ii} & \vdots & \tilde{B}_i \\ \cdots & \cdots & \cdots \\ \tilde{C}_i & \vdots & \widetilde{D}_i \end{bmatrix}$$

for $i = 1, 2$. Note that the dimensions of realizations of $G_1(s)$ and $G_2(s)$ are equal to the dimensions of M and N, respectively. If we begin with a minimal realization of $G(s)$, then the realizations of $G_1(s)$ and $G_2(s)$ are necessarily minimal. The factorization is also minimal since the McMillan degrees of $G_1(s)$ is the sum of McMillan degrees of $G_1(s)$ and $G_2(s)$.

5 J-Inner-Outer Factorization

The J-inner-outer factorization problem arises in H^∞ optimization [1, 10]. All controllers which stabilize a given physical plant and make the H^∞ norm of the closed loop transfer function less than or equal to a prespecified constant can be parametrized via J-inner-outer factorizations. Take for instance the problem of weighted sensitivity minimization

[10]. Given a model $G(s)$ of a physical process and weighting matrices $W_1(s)$ and $W_2(s)$, we seek to characterize all controllers $K(s)$ which stabilize $G(s)$ and satisfy

$$\|W_1(s)(I+G(s)K(s))^{-1}W_2(s)\|_\infty \leq 1 \tag{5.1}$$

If $W_2(s)$ is invertible, we form the augmented plant

$$\tilde{P}(s) = \begin{bmatrix} 0 & W_1 \\ W_2^{-1}G & W_2^{-1} \end{bmatrix} \tag{5.2}$$

and perform a factorization

$$\tilde{P}(s) = \Theta(s)R(s) \tag{5.3}$$

such that $\Theta(s)$ is J-inner and $R(s)$ 9s outer. A rational matrix $\Theta(s)$ is J-inner if

$$\Theta^*(j\omega)J\Theta(j\omega) = J,$$

where $\Theta^*(j\omega)$ represents the complex conjugate transpose of $\Theta(j\omega)$, and

$$\Theta^*(s)J\Theta(s) = J.$$

In the above,

$$J = \begin{bmatrix} I & 0 \\ 0 & -I \end{bmatrix}$$

A rational matrix $R(s)$ is outer if $R(s) \in H^\infty$ and $R^{-1}(s) \in H^\infty$. Let

$$\Omega(s) \triangleq R^{-1}(s) = \begin{bmatrix} \Omega_{11} & \Omega_{12} \\ \omega_{21} & \Omega_{22}(s) \end{bmatrix}$$

The set of all controllers which stabilize $G(s)$ and satisfy (5.1) is now parametrized by

$$K(s) = (\Omega_{11}U + \Omega_{12})(\Omega_{21}U + \Omega_{22})^{-1} \tag{5.4}$$

where $U(s) \in H^\infty$ and $\|U(s)\|_\infty \leq 1$.

Given $\tilde{P}(s)$, it is thus of interest to compute $R(s)$ and $R^{-1}(s)$. Let $\tilde{P}(s)$ have the minimal realization

$$\tilde{P}(s) = \begin{bmatrix} A & \vdots & B \\ \cdots & \cdots & \cdots \\ C & \vdots & D \end{bmatrix}$$

Form

$$V(s) = \tilde{P}^*(s)J\tilde{P}(s)$$

which has the realization

$$V(s) = \begin{bmatrix} -A^* & C^*JC & \vdots & C^*JD \\ 0 & A & \vdots & B \\ \cdots & \cdots & \cdots & \cdots \\ -B^* & D^*JC & \vdots & D^*JD \end{bmatrix} \tag{5.5}$$

From (5.3) it follows that

$$V(s) = \tilde{P}^*(s)J\tilde{P}(s) = R^*(s)JR(s) \tag{5.6}$$

since

$$\Theta^*(s)J\Theta(s) = J \tag{5.7}$$

Let us assume, for the sake of simplicity, that $\widetilde{P}(s)$ is square. Thus, (5.7) and the inverstibility of $R(s)$ imply that D is invertible. Also, a realization of $V^{-1}(s)$ is found to be

$$V^{-1}(s) = \left[\begin{array}{cccc} -(A^x)^* & 0 & \vdots & C^*(D^*)^{-1} \\ BEB^* & A^x & \vdots & BE \\ \cdots & \cdots & \cdots & \cdots \\ EB^* & -D^{-1}C & \vdots & \mathrm{E} \end{array}\right] \tag{5.8}$$

where

$$A^x = A - BD^{-1}C$$
$$\mathrm{E} = (D^*JD)^{-1}$$

We form now the Hamiltonian matrices

$$H_1 = \begin{bmatrix} -A^* & C^*JC \\ 0 & A \end{bmatrix}$$

and

$$H_2 = \begin{bmatrix} -(A^*)^* & 0 \\ BEB^* & A^x \end{bmatrix}$$

Since $R(s) \in H^\infty$ and $R^{-1}(s) \in H^\infty, V(s)$ in (5.6) has no poles or zeros on $s = j\omega$. Thus H_1 and H_2 have no imaginary eigenvalues. Hence, there exist Hermitian solutions $X = X^*$ and $Y = Y^*$ satisfying the algebraic Riccati equations

$$XA^x + (A^x)^*X + XBEB^*X = 0 \tag{5.9}$$
$$YA^* + AY - YC^*JCY = 0 \tag{5.10}$$

with the corresponding stabilizing side conditions

$$\lambda(A^* + BEB^*X) \subset Re(s) < 0 \tag{5.11}$$

and

$$\lambda(A - YC^*JC) \subset Re(s) < 0 \tag{5.12}$$

The invariant subspaces are picked to be of the form

$$M = I_m \begin{bmatrix} I \\ X \end{bmatrix}, N = I_m \begin{bmatrix} Y \\ I \end{bmatrix}$$

which form a direct sum decomposition of $\mathbb{C}^n$ if and only if

$$T = \begin{bmatrix} I & Y \\ X & I \end{bmatrix}$$

is invertible. This holds if and only if $(I - XY)$ is invertible in which case we define

$$Z = (I - XY)^{-1}$$

Assuming the existence of Z, th minimal factorization procedure from section 4 can be carried out leading to a realization of $R(s)$ given by

$$R(s) = \left[\begin{array}{ccc} A - YC^*JC & \vdots & B + YC^*JD \\ \cdots & \cdots & \cdots \\ DE(D^*JC + B^*X)Z^* & \vdots & D \end{array} \right] \tag{5.13}$$

A realization of $\Theta(s)$ is now computed via

$$\Theta(s) = \tilde{P}(s)R^{-1}(s)$$

which is J-inner if and only if $X \geq 0, Y \geq 0$ and $\lambda_{\max}(XY) < 1$.

6 Concluding Remarks

The factorization results of sections 3 - 5 are derived based on the assumption that the invariant subspaces M and have bases which can be constructed from solutions to Riccati equations. The extent to which this assumption holds in specific cases needs to be clearly established. For the remaining cases, factorization techniques need to be developed which rely on the general similarity transformation matrix T in which T_{11} and T_{22} are not necessarily invertible. When the subspaces M and N form a direct sum decomposition of $\mathbb{C}^n$, abstract formulae for the factors $G_1(s)$ and $G_2(s)$ in terms of projections on M and N are derived in [4]. Good computational schemes for computing these projections need to be developed. In particular, solutions to the J-inner-outer factorization problem in terms of the projections and good computational schemes for implementing the solution would be of interest.

REFERENCES

[1] Ball, J.A., J.W. Helton and M Verma, "A Factorization Principle for Stabilization of Linear Control Systems", Int. J. Robust and Nonlinear Control, 1, 229-294, 1991.

[2] Ball, J.A. and A.C.M. Ran, "Optimal Hankel Norm Model Reductions and Wiener-Hopf actorization I: The Canonical case", SIAM J. Control and Optimization, 25, 362-382, March 1987.

[3] Bast, H., I Grohberg and M.A. Kaashoek, Minimal Factorization of Matrix and Operator Functions, Birkhauser, Basel, 1979.

[4] Bast, H., I Grohberg, M.A. Kaashoek and P. VanDoorer, "Factorizations off Transfer Function", SIAM J. Control and Optimization, 18, 675-696, 1980.

[5] Chen, C.T., Linear System Theory and Design, HRW, 1984.

[6] Laub, A.J., "A Schur Method for Solving Algebraic Riccati Equations", IEEE Trans. Automat. Contr., 24, 913-921, 1979.

[7] Verma, M., "Robust Stabilization of Linear Time-Invariant Systems", IEEE Trans. Automat. Contr., 34, 872-876, 1989.

[8] Vidyasagar, M., Control System Synthesis - A Factorization Approach, MIT Press, Cambridge, MA, 1985.

[9] Youla, D.C., H.A. Jaby and J.J. Bongiorno Jr., "Modern Wiener-Hopt Design of Optimal Controllers: part II", IEEE Trans. Automat. Contr., 21, 319-338, 1976.

[10] Zames, G., "Feedback and Optimal Sensitivity: Model Reference Transformations, Multiplicative Seminorms and Approximate Inverses", IEEE Trans. Automat. Contr, 26, 301-320, 1981.

MODEL REDUCTION VIA STATE SPACE PERTURBATIONS

Herbert W. Beke
Dept. of Electrical Engineering
University of Minnesota
Minneapolis, MN. 55455
beke@ee.umn.edu

Daniel Boley
Dept. of Computer Science
University of Minnesota
Minneapolis, MN. 55455
boley@cs.umn.edu

Presented at the Thirty-first Annual Allerton Conference on Communications, Control, and Computing, at Allerton House, University of Illinois at Urbana-Champaign. 9/29–10/1, 1993.

Abstract. Given a large order realization of a linear dynamical system, we find a lower order model obtained by applying small perturbations to the original model. We show how the small perturbations may be computed and how to compute the lower order model using only orthogonal similarity transformations. We illustrate this method by comparing its performance with the truncated balanced realization and the optimal Hankel norm approximate. We find that the phase is tracked best by our method on the examples we tried.

1. Introduction

Model reduction refers to the approximation of a linear system model by a lower order model such that certain criteria are met. The motivations for model reduction are mainly economical since low order models can be analyzed, simulated or built at a much lower cost than high order models. Numerous papers have been written on model reduction. Reduction by modal expansion appears to be one of the oldest approaches. However, reduction by balanced realization [7] and by optimal Hankel norm approximation [9] as well as their weighted versions [8,10] have now found widespread acceptance among control engineers. In this paper, we present a novel constructive way of implementing model reduction based on some recent results in matrix pencil theory [1].

We now state some definitions and characterizations of matrix pencils [1,3,4,5]. Let A and B be two n x p matrices. The set of all matrices of the form $(A - \lambda B)$, where λ is any complex number is said to be a matrix pencil of dimension n x p. Two matrix pencils $(A_1 - \lambda B_1)$ and $(A_2 - \lambda B_2)$ of dimension n x p are said to be strictly equivalent when there exist constant invertible matrices P and Q of dimension n x n and p x p respectively such that $P(A_1 - \lambda B_1)Q = (A_2 - \lambda B_2)$. In the classical theory of matrix pencils, it is well known that any pencil is strictly equivalent to its Kronecker Canonical Form (KCF). A matrix pencil $P(\lambda) = (A - \lambda B)$ is said to be deficient if there exists some λ for which it is not full rank. If $P(\lambda) = (A - \lambda B)$ is always full rank for any value of λ, then it is said to be non–deficient.

We shall consider the standard finite dimensional linear time invariant (FDLTI) continuous time system described by

$$\dot{x} = Fx + Gu; \quad y = Hx, \tag{1}$$

where $F \in R^{p \times p}$, $G \in R^{p \times m}$, and $H \in R^{r \times p}$ as the full order model. F, G, and H are also known as the state, input, and output matrices respectively. We have omitted the feedthrough matrix in the state space model description because it remains invariant in the model reduction scheme proposed in this paper. If the full order model is not strictly proper, the method and analysis outlined in this paper can still be applied to the strictly proper part of the model. We shall assume that the system described by (1) is stable, i.e., the real part of all the eigenvalues of F is strictly less than zero. We want to remark that this is a very mild assumption because the transfer function matrix $G(s)$ of any FDLTI system can be decomposed as $G(s) = G_S(s) + G_U(s)$, where $G_S(s)$ is the transfer function matrix of the stable part and $G_U(s)$ is the transfer function matrix of the unstable part. One can then proceed with the model order reduction of the stable part. We shall also assume that the state space realization described by (1) is minimal. According to the PBH rank test (see e.g., [4]), the dynamical system described by (1) is controllable if and only if the matrix pencil $P^T(\lambda) = [F-\lambda I|G] = [F|G] - \lambda[I|0]$, has full rank for any complex value λ.

In [1], the sensitivity of an algebraic (Kronecker) structure of rectangular matrix pencils to perturbations in the coefficients was examined and eigenvalue perturbation bounds in the spirit of Bauer-Fike was used to develop computational upper and lower bounds on the distance from a given pencil to one with a qualitatively different Kronecker structure. In this paper, we exploit the bounds derived in [1] for estimating the upper bound on the distance to a deficient pencil [1], to implement a model reduction scheme.

The rest of the paper is organized as follows. In Section 2, we review the background material on the computation of the nearest distance to a deficient pencil as espoused in [1]. In Section 3, we discuss how the ideas in Section 2 can be employed in a model reduction. In Section 4 we present an algorithm for model reduction and make some remarks on how it can be modified to suit the particular needs of the designer. In Section 5 we address the issue of the stability of the reduced order model. In Section 6 we derive the state space expression for the model reduction error in terms of the full order model i.e., (1) and the perturbation of the matrix pencil that leads to a deficient pencil. For a special case of our method, we show in Section 5 that the reduced order model is stable so long as the full order model is stable; and in Section 6, we derive an upper bound on the H_2 norm of the model reduction error for that special case. In Section 7, we present some numerical results. Finally, in Section 8, we make some concluding remarks.

2. Computation of Deficient Pencil

In this section we discuss two issues of importance to the reduction scheme as they were espoused in [1]. The first one concerns the problem of determining whether a given rectangular pencil is deficient or not. Specifically, given an n x p pencil $P(\lambda) = (A - \lambda B)$, with $n > p$, determine whether or not $(A - \lambda B)$ loses rank for any λ, including possibly λ infinite. By augmenting A and B matrices with arbitrary n x $(n - p)$ matrices C, D. We can examine the square n x n generalized eigenvalue problem

$$[A,C]v = \lambda[B,D]v \ . \tag{2}$$

Now, by partitioning the vector v as $v^T \equiv [x^T,y^T]$, where x is a p-vector, and y is a $(n-p)$-vector, the following proposition can be easily derived.

Proposition 1 [Boley]: Given an n x p pencil $(A - \lambda B)$ with $n > p$, and given an arbitrary full–rank n x $(n - p)$ matrices C, D, the following are equivalent:

(a) $A - \lambda B$ is a deficient pencil.

(b) Equation (2) has an annihilating vector v_0 whose last $n - p$ components y_0 are zero. Call the corresponding annihilating value λ_0.

Furthermore, we have the following:

(c) If B has full column rank, then all the annihilating vectors v_0 and corresponding values λ_0 are exactly the generalized eigenpairs for the regular part of the pencil.

Proof: See [1].

■

The issue of choosing C and D has also been addressed in [1] where two choices were proposed for the selection of C and D. One of these choices is to select C and D as orthonormal basis of the space orthogonal to the columns of A and B. This choice has the effect of limiting the increase to the condition numbers of $[A,C]$ and $[B,D]$ with respect to inversion. Thus in the case $B = [I_p,0]^T$, D is chosen as $D = [0,I_{n-p}]^T$ to turn the problem into an ordinary eigenvalue problem.

The second issue concerns the problem of computing the upper bound on the distance to a deficient pencil. Specifically, consider a nondeficient n x p pencil $P(\lambda) = (A - \lambda B)$. In this case, we know that B has full rank. We would like to estimate the size of the perturbation E to the matrix A that is needed to obtain a deficient pencil $(A + E - \lambda B)$. In [2], it was shown that the smallest perturbation E can be obtained by solving the minimization problem

$$\min_s \sigma_{\min}(A - sB) \; , \tag{3}$$

where $\sigma_{\min}(M)$ denotes the smallest singular value of the matrix M, and s varies over the entire complex plane. If we denote by σ^* and s^* the minimum in (3) and value of s achieving that minimum, respectively, then $\|E\| = \sigma^*$. In [1], it was shown that a simpler scheme which provides both a good estimate for $\|E\|$ and for that value of s that yields the minimum in (3) involves solving the eigenvalue problem (2). Let

$$\lambda_i,\ v_i := [x_i^T, y_i^T]^T, \qquad i = 1,\cdots,n \tag{4}$$

be the generalized eigenvalues and eigenvectors for (2). For each i we have the equation $[A,C]v_i = \lambda_i[B,D]v_i$, using (4) this equation can be rewritten as $(\lambda_i B - A)x_i = (C - \lambda_i D)y_i$. For each i define the residual r_i as $r_i := (A - \lambda_i B)x_i$; and the perturbation E_i as

$$E_i := \frac{-r_i x_i^T}{\|x_i\|^2} \equiv (\lambda_i B - A)\frac{x_i x_i^T}{\|x_i\|^2} = (C - \lambda_i D)\frac{y_i}{\|x_i\|}\cdot\frac{x_i^T}{\|x_i\|} \; . \tag{5}$$

We remark that the norm on E_i can be simply computed as $\|E_i\| = \|r_i\|_2/\|x_i\|_2$. Note that $E_i x_i = -r_i$, and thus the expression for the residual becomes $(A + E_i - \lambda_i B)x_i = 0$. Now, $(A + E_i - \lambda B)$ is a deficient pencil, losing rank exactly at $\lambda = \lambda_i$, for each i. Let σ_i, u_i, w_i be, respectively, the smallest singular value and the corresponding left and right singular vectors of $(A - \lambda_i B)$ for each i. Then $E_i' := -\sigma_i u_i w_i^T$ is another smaller perturbation yielding a deficient pencil. By taking norms of (5), the following bounds are obtained for the perturbations: $\|E_i'\| \le \|E_i\| \le \|(C - \lambda_i D)y_i\|/\|x_i\|$. We also have the following error bound from [1]. If E denotes that perturbation with the smallest norm yielding a deficient pencil, then E satisfies

$$\|E\| \equiv \sigma^* \le \beta_2 \equiv \min_i \|E_i'\| \equiv \min_i \sigma_{\min}(A - \lambda_i B) \le \beta_1 \equiv \min_i \|E_i\| \equiv \min_i \frac{\|(C - \lambda_i D)y_i\|}{\|x_i\|} \; . \tag{6}$$

The importance of this error bound was noted and stated as the following proposition in [1].

Proposition 2 [Boley]: Let $(A - \lambda B)$ be an n x p pencil, with $n > p$. Let C, D be two arbitrary full-rank n x $(n - p)$ matrices. Then the smallest perturbation E such that $A+E-\lambda B$ is a deficient pencil satisfies the bound (6), where λ_i, v_i, $i = 1,\cdots,n$ are the eigenpairs of the generalized eigenproblem (2) and y_i are defined by (4).

Proof. See [1].

■

3. Application to Model Reduction

Our goal is to use the results of the previous section to implement a model reduction scheme. This is done by constructing a matrix pencil based on the PBH test for controllability or observability. The matrix pencil so constructed is augmented with "C" and "D" matrices as described in Section 2. One then proceeds to solve the eigenvalue problem which is (2). Next, the norm of the perturbations $\|E_i\|$, $i,\cdots,n$, are computed, and sorted in increasing order. Using the eigenpair (λ_i,v_i) corresponding to the smallest value of $\|E_i\|$, one constructs a perturbation matrix E_i such that $(A+E_i-\lambda B)$ is a deficient pencil, losing rank exactly at $\lambda=\lambda_i$. The resulting uncontrollable or unobservable state is then truncated to yield a reduced order model.

3.1 Reduction by a Single Mode Using Controllability Criteria

Here, we shall assume that the A and B matrices of the matrix pencil $P(\lambda) = (A - \lambda B)$ are defined by: $A = [F,G]^T$, $B = [I_p,0]^T$, where the underlying matrices F and G are given by (1). Suppose the perturbation matrix E_i from (6) is partitioned as $E_i^T = [E_1,E_2]$, where E_1 and E_2 have the dimensions of F and G respectively. Substituting for the A, B, and E_i matrices in the expression for the residual i.e., $(A + E_i - \lambda_i B)x_i = 0$, we obtain:

$$\left[\begin{pmatrix} F^T+E_1^T \\ G^T+E_2^T \end{pmatrix} - \lambda_i \begin{pmatrix} I_p \\ 0 \end{pmatrix}\right] x_i = 0 . \tag{7}$$

We now look at the two set of equations in (7). The top part yields $(F + E_1)^T x_i = \lambda_i x_i$. When this is premultiplied by a Householder transformation Q such that $Qx_i = e_1$, where e_1 is the first column of the identity matrix, we obtain $Q(F + E_1)^T Q^T e_1 = \lambda_i Q x_i = \lambda_i e_1$. The bottom part of (7) yields $(G + E_2)^T x_i = 0$. When this is transposed, we obtain $x_i^T Q^T Q(G+E_2) = e_1^T Q(G+E_2) = 0$. The perturbed system can be written as (8) where $Q(F + E_1)Q^T$ and $Q(G + E_2)$ have the structures shown in (9).

$$\dot{z} = Q(F+E_1)Q^T z + Q(G+E_2)u \; ; \quad y = HQ^T z , \tag{8}$$

$$Q(F+E_1)Q^T = \begin{bmatrix} \lambda_i & 0 & \cdots & 0 \\ x & x & \cdots & x \\ \vdots & \vdots & \ddots & \vdots \\ x & x & \cdots & x \end{bmatrix}, \qquad Q(G+E_2) = \begin{bmatrix} 0 & \cdots & 0 \\ x & \cdots & x \\ \vdots & \ddots & \vdots \\ x & \cdots & x \end{bmatrix}. \tag{9}$$

Clearly, the first state in the transformed state z is not controllable. If we denote $Q(F+E_1)Q^T$, $Q(G + E_2)$, HQ^T by $\tilde{F}$, $\tilde{G}$, and $\tilde{H}$ respectively, the perturbed system described by (8) can be written in partitioned form as

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} \bar{F}_{11} & 0 \\ \bar{F}_{21} & \bar{F}_{22} \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \bar{G}_2 \end{bmatrix} u \; ; \; y = \begin{bmatrix} \bar{H}_1 & \bar{H}_2 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} . \tag{10}$$

If the uncontrollable state is stable it can be directly truncated to yield a reduced order model:

$$\dot{z}_2 = \bar{F}_{22} z_2 + \bar{G}_2 u \; ; \; y = \bar{H}_2 z_2 \; . \tag{11}$$

3.2 Reduction by a Single Mode Using Observability Criteria

One can also use observability criteria by doing exactly the same reduction on the dual system: $\dot{x} = F^T x + H^T u; \ y = G^T x$.

3.3 On the Choice of Controllability or Observability Criteria

The trace of the controllability and observability grammians give a measure of how "controllable" or how "observable" the model is in its current state space realization. Denote by κ_C the norm of the perturbation to the input matrix G that will render the realization uncontrollable, and κ_O the norm of the perturbation to the output matrix H that will render the same realization unobservable. We shall also denote the controllability and observability grammians by W_{contr} and W_{obs} respectively. It is desirable to effect the model reduction by applying the smallest possible perturbation. If trace(W_{contr}) < trace(W_{obs}) then we normally expect to have $\kappa_C < \kappa_O$, in that case the controllability method should be used. On the other hand if trace(W_{contr}) > trace(W_{obs}) then we normally expect to have $\kappa_C > \kappa_O$, and in that case the observability method should be used.

4. Algorithm for Model Reduction

Given a FDLTI system as described by (1), a reduced order model can be computed using the following algorithm.

Step 1: Compute trace(W_{contr}) and trace(W_{obs}). If trace(W_{contr}) < trace(W_{obs}) then use the PBH controllability test to form the matrix pencil $P(\lambda) = (A - \lambda B)$ where A and B are defined by $A = [F,G]^T$, $B = [I_p,0]^T$. Otherwise use the PBH observability test to form the matrix pencil $P(\lambda) = (A - \lambda B)$ with A and B defined by $A = [F^T,H^T]^T$, $B = [I_p,0]^T$. Augment the pencil with C and D matrices as described in Section 2.

Step 2: Solve the corresponding eigenvalue problem which is (2). Next, compute $\|E_i\|$, $i = 1,\cdots,n$, and sort them in increasing order.

Step 3: Using the eigenpair corresponding to the smallest value of $\|E_i\|$, construct the perturbation matrix E_i using (5). Next form the perturbed system as described in Section 3 and then apply the appropriate Householder transformation Q to the perturbed system to reveal the uncontrollable or unobservable state. Verify the stability of the perturbed system. A reduced order model can be obtained by direct truncation if the resulting uncontrollable or unobservable state is stable.

Remarks:

1. The algorithm proposed above could be extended to include frequency dependent weightings. The modes of the full order model can be weighted differently and a different criterion can be used for truncation. For example, one could use a weighted average of the existing measure and the maximum singular values of the transfer

function matrix. For simplicity, we have not done that here.

2. The method presented in this paper involves using a Householder transformation Q to effect a similarity transformation. Model reduction by optimal Hankel norm reduction [9] and by truncation by balanced realization [7] also involve using a matrix T to effect a similarity transformation, however, the T used in [7,9] is not in general an orthonormal matrix. As is well known, use of non-orthonormal (or non-unitary) transformation can lead to worsening of the condition numbers and loss of accuracy of the system matrices. Because of this fact, the method presented in this paper is at a significant advantage over the methods of [7,9].

3. In [14], we show how to extend the method presented in this paper so that one can reduce the order of the model by multiple states instead of a single state each time the matrix pencil is perturbed.

5. Stability of the Reduced Order Model

A question of great importance in any model reduction scheme is whether the reduced order model obtained from a stable full order model is stable or not. In our scheme, this amounts to asking whether the real part of all the eigenvalues of $Q(F+E_1)Q^T$ are strictly less than zero. First of all we note that $(F+E_1)$ and $Q(F+E_1)Q^T$ are similar matrices; therefore they have the same eigenvalues with the same multiplicities. Consequently, it will suffice to consider the eigenvalues of $(F+E_1)$. We shall focus on the controllability condition, however the discussions concerning the stability of $(F+E_1)$ are equally valid for the observability condition.

In order to form the perturbation matrix E_i, we solve the eigenvalue equation $[A,C]v_i = \lambda_i[B,D]v_i$, where A is formed as $A = [F,G]^T$, $B = [I_p,0]^T$ and set $D = [0,I_{n-p}]^T$ to turn the problem into an ordinary eigenvalue problem. We also select C as orthonormal basis of the space orthogonal to the columns of A. Furthermore, we partition the matrix C as $C^T = [C_1^T,C_2^T]$, where C_1 and C_2 are of dimensions p x $(n-p)$ and $(n-p)$ x $(n-p)$ respectively. We also introduce the parameter α by writing C as $C^T = [\alpha C_1^T,C_2^T]$, with $\alpha \in [0,1]$. When $\alpha = 0$, there is no perturbation of the F matrix. In that case the reduced order model remains stable so long as the original model is stable. As α increases, the perturbation E_1 to F is no longer zero but varies continuously with α. Therefore stability of $(F + E_1)$ is maintained for small values of α. Application of the Bauer-Fike theorem [6] gives us the following result. Let K be the condition number of the matrix of eigenvectors of F, and d the distance to the imaginary axis of the eigenvalue of F which is closest to the imaginary axis. Then stability of $(F + E_1)$ is guaranteed as long as $K\|E_1\| < d$. The reader is referred to [14] for details. In practice we found it just sufficed to verify the stability of the perturbed system.

5.3 Spurious Eigenvalues

Another issue of concern is that of the $n - p$ spurious eigenvalues introduced when the eigenvalue equation is solved. These spurious eigenvalues arise because the underlying dynamical system of the matrix pencil is of order p whereas the eigenvalue equation yields n eigenvalues. Note that when $\alpha = 0$, the spurious eigenvalues pose no problem because the vector $x_i = 0$, thus $\|E_i\| = \infty$ for the spurious eigenvalues and hence can be ignored in the model reduction process. However, when $\alpha \neq 0$ the value of $\|E_i\|$ for the spurious roots is finite; but generally, $\|E_i\|$ for the spurious eigenvalues will be much larger than for the non-spurious eigenvalues. Space does not permit an extensive discussion here, but the reader is referred to [14] for a detailed explanation.

6. Model Reduction Error

In this section we derive state space expressions for the model reduction error. We shall assume that the reduced order model was obtained using the controllability condition. By a slight modifications of the equations below one can obtain similar expressions for a reduced order model obtained using the observability condition. We consider two cases which are 1) $\alpha = 0$, and 2) $\alpha \neq 0$.

When $\alpha = 0$, then E_1 is a zero matrix which implies there is no perturbation of the F matrix and it is only the G matrix which is perturbed by E_2. Suppose we apply a similarity transformation to the state space representation of the full order model $G(s)$ (i.e., (1)), using the Householder transformation Q from Section 3.1. Denote QFQ^T, and HQ^T by $\hat{F}$ and $\hat{H}$ respectively. Then it can be shown that the model reduction error $K(s) = G(s) - G_r(s)$ is given by:

$$\dot{z} = \hat{F}z + (-QE_2)v \ ; \quad y = \hat{H}z \ . \tag{12}$$

We now seek to compute an upper bound on the H_2 norm of the model error $K(s)$ when $\alpha = 0$. To simplify the notation we shall denote $-QE_2$ by E_C. Using equation (5.35) from [11] to compute the H_2 norm of $G(s)$ and $K(s)$ we have:

$$\|G\|_2^2 = \mathrm{trace}\,(\Gamma^T W_{obs} \Gamma) \ ; \quad \|K\|_2^2 = \mathrm{trace}\,(E_C^T W_{obs} E_C) \ , \tag{13}$$

where W_{obs} is the observability grammian of both $G(s)$ and $K(s)$. Let $W_{obs} = LL^T$ be the cholesky factors for W_{obs}, a positive definite symmetric matrix with eigenvalues we will call $\sigma_1^2, \cdots, \sigma_p^2$. Then $\mathrm{trace}(W_{obs}) = \sigma_1^2 + \cdots + \sigma_p^2 = \|L\|_F^2$. In general, $\mathrm{trace}(M^T M) \equiv \|M\|_F^2$. Then we have

$$\mathrm{trace}\,(E_C^T W_{obs} E_C) = \mathrm{trace}(E_C^T L L^T E_C) = \|L^T E_C\|_F^2 \leq \|L\|_F^2 \|E_C\|_F^2 = \mathrm{trace}(W_{obs}) \cdot \|E_C\|_F^2 \ . \tag{14}$$

Now $\|E_C\|_F$ is small since $E_C = -QE_2$, and E_2 is just the perturbation needed in order to arrive at the uncontrolled system.

Here is an "analysis" for the term $\mathrm{trace}(W_{obs})$. Roughly, this term measures the energy that can be retrieved at the output from the system states, and $\|G\|_2^2$ measures the energy that can be retrieved at the output from a unit impulse input. We may define the ratios:

$$\rho_{input} = \frac{\sqrt{\mathrm{trace}\,(W_{obs})}}{\|G\|} \ ; \quad \rho_{output} = \frac{\sqrt{\mathrm{trace}\,(W_{contr})}}{\|G\|} \ . \tag{15}$$

where ρ_{input} roughly measures the *input transmission ratio* of energy, and ρ_{output} roughly measures the *output transmission ratio* of energy. These ratios are properties inherent to a given state-space realization, regardless of whatever model reduction method is used.

Using the ratio ρ_{input} defined in (15), we may bound the norm of the error transfer function $K(s)$ as $\|K\| \leq \rho_{input} \|G\| \|E_C\|_F$. We have computed the reduced order system by applying a controllability perturbation to the inputs. We could also apply an observability perturbation to the outputs, obtaining a reduced order model $\hat{G}_r(s)$ and error transfer function $K_{obs}(s) = G(s) - \hat{G}_r(s)$. Working through the same development, we could obtain the bound $\|K_{obs}\| \leq \rho_{output} \|G\| \|E_O\|_F$; where $E_O = -E_2 Q^T$, and E_2 is the perturbation to the output matrix H.

Note that $\rho_{input} > \rho_{output}$ is equivalent to trace(W_{contr}) < trace(W_{obs}). As discussed in Section 3.3, the trace of the controllability and observability grammians give a criteria for deciding whether to effect model reduction by input or output perturbation. Because of the equivalence relationship between the ratios and the trace of the grammians, the criteria for deciding which method to choose can be stated as follows: if $\rho_{input} > \rho_{output}$ choose the input perturbation method otherwise choose the output perturbation method.

When $\alpha \neq 0$ then in general the eigenvalues of the full order model are different from the eigenvalues of the reduced order model and we do not have any simple way of expressing the model error dynamics. However we can augment the states x of the full order model with the states z_2 of the reduced order model to form a composite system with states (x,z_2). From (1) and (11) we form the state space representation for the model reduction error $K(s)$ as

$$\begin{bmatrix} \dot{x} \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} F & 0 \\ 0 & \bar{F}_{22} \end{bmatrix} \begin{bmatrix} x \\ z_2 \end{bmatrix} + \begin{bmatrix} G \\ \bar{G}_2 \end{bmatrix} u \; ; \quad y_e = \begin{bmatrix} H & -\bar{H}_2 \end{bmatrix} \begin{bmatrix} x \\ z_2 \end{bmatrix} . \tag{16}$$

For both $\alpha = 0$ and $\alpha \neq 0$, the state space expression for the model reduction error is in terms of the full order model, the perturbation E_i and an orthonormal matrix Q. The sup-norm of the model reduction error can be computed using an iteration method such as [12].

7. Numerical Results

The example is the "JPL 8-state model" taken from page 32 of [15], which is a model of SISO large flexible structure. We applied four times the matrix pencil reduction scheme described in Section 3.1; the "D" and "C" matrices were chosen as outlined in Section 5 with $\alpha = 1$. Figure 1.1 is the Bode magnitude and phase plots of the full order and the reduced order 4 model using the matrix pencil approach. We next compared the performance of the matrix pencil reduction scheme with 1) truncation by balanced realization [7], and 2) optimal Hankel norm reduction [9] methods. The balanced realization and the optimal Hankel norm reduced order models were obtained by using functions "strunc" and "hankmr" from [13]. Figure 1.2 shows the Bode plots of the full order and the reduced order 4 models obtained from balanced realization and optimal Hankel norm methods.

From Figures 1.1 and 1.2, we see that both the matrix pencil and the balanced realization methods yield a reduced order 4 model whose Bode plots fit the full order model well at low and intermediate frequencies; the optimal Hankel norm approach gives the worst fit. The plots also show that the phase error over the entire frequency range is least for the model obtained from the matrix pencil approach.

8. Conclusion

In this paper we have presented a novel way of model reduction based on matrix pencil theory. We have given some initial results on the stability of the reduced order model in terms of a perturbation applied to the system matrix F. However, for the special case of $\alpha = 0$, the reduced order model is always stable so long as the full order model is. We have also derived an upper bound on the H_2 norm of the model reduction error when $\alpha = 0$. The computational cost of the matrix pencil method is comparable to other methods which require balanced realization. Some of the advantages of the matrix pencil method over the balanced realization and optimal Hankel norm methods are: 1) there is no need to transform the full order model into balanced realization, and 2) the matrix pencil method uses only orthonormal transformations. We have given a detailed example to illustrate features of the method. The results indicate that the matrix pencil yields models that are much better approximations than those from the optimal Hankel norm method. The models from the matrix pencil method tend to follow the phase of the full order model better than models from the balanced realization method, and are otherwise comparable.

REFERENCES

[1] D. Boley, *Estimating the Sensitivity of the Algebraic Structure of Pencils with Simple Eigenvalue Estimates*, SIAM J. Matrix Anal. Appl., Vol. 11, No. 4, Oct. 1990, pp. 632-643.

[2] D. Boley, *Computing Rank-Deficiency of Rectangular Matrix Pencils*, Systems Control Lett., 9 (1986), pp. 207-214.

[3] P. Van Dooren, *The Computation of Kronecker's Canonical Form of a Singular Pencil*, Linear Algebra Appl., 27 (1979) pp. 103-141.

[4] T. Kailath, *Linear Systems*, Prentice–Hall, Englewood Cliffs, NJ, 1980.

[5] F. R. Gantmakher, *Theory of Matrices*, Vols. 1 and 2, Chelsea, New York, 1959.

[6] G. H. Golub and C. Van Loan, *Matrix Computations*, The John Hopkins University Press, Baltimore, MD, 1983.

[7] B. C. Moore, "Principal component analysis in linear systems: controllability, observability, and model reduction," *IEEE Trans. Auto. Control*, Vol. AC-26, No. 1, 1981.

[8] D. Enns, "Model reduction for control system design," Ph.D. dissertation, Dept. of Aeronautics and Astronautics, Stanford University, June 1984.

[9] K. Glover, "All optimal Hankel-norm approximations of linear multivariable systems and their L_∞-error bounds," *IJC*, 1984, Vol 39, No. 6, 1115-1193.

[10] G.O. Lathan and B.D.O. Anderson, "Frequency-Weighted optimal Hankel-norm approximation of stable transfer functions," *Systems & Control Letters* 5 (1985) 229-236.

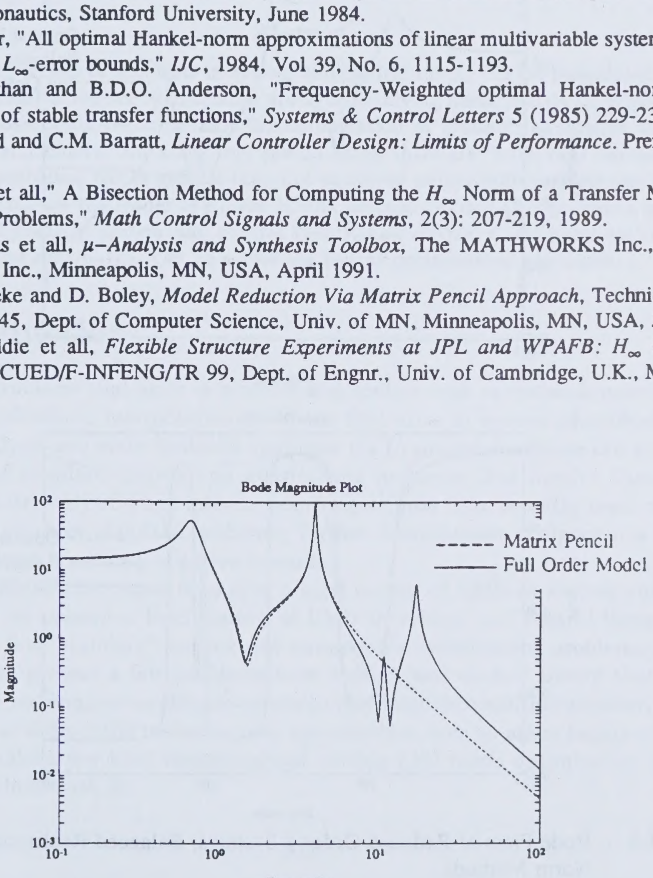
[11] S.P. Boyd and C.M. Barratt, *Linear Controller Design: Limits of Performance*. Prentice-Hall, 1991.

[12] S. Boyd et all," A Bisection Method for Computing the H_∞ Norm of a Transfer Matrix and Related Problems," *Math Control Signals and Systems*, 2(3): 207-219, 1989.

[13] G.J. Balas et all, *μ–Analysis and Synthesis Toolbox*, The MATHWORKS Inc., MA. and MUSYN Inc., Minneapolis, MN, USA, April 1991.

[14] H. W. Beke and D. Boley, *Model Reduction Via Matrix Pencil Approach*, Technical Report # TR 93-45, Dept. of Computer Science, Univ. of MN, Minneapolis, MN, USA, June 1993

[15] S.A. Buddie et all, *Flexible Structure Experiments at JPL and WPAFB: H_∞ Controller Designs*, CUED/F-INFENG/TR 99, Dept. of Engnr., Univ. of Cambridge, U.K., May 1992.



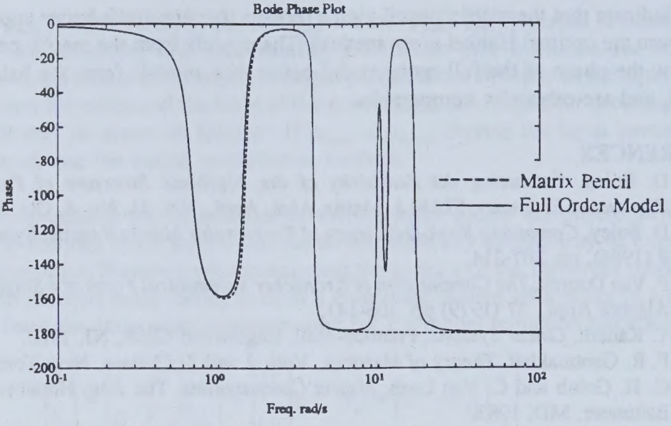


Figure 1.1 Bode Plots of Reduced Order 4 System; Matrix Pencil Approach.

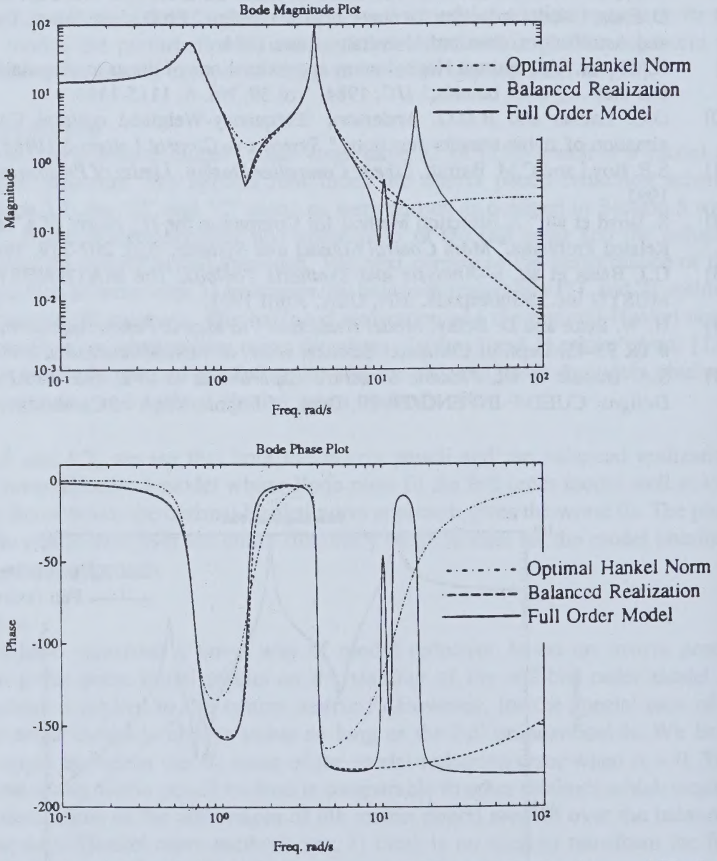


Figure 1.2 Bode Plots of Reduced Order 4 Systems; Balanced Realization and Hankel Norm Methods.

Linear Matrix Inequalities in System and Control Theory

S. Boyd[*] L. El Ghaoui[‡] E. Feron[¶] and V. Balakrishnan[**]

[*]Associate Professor of Electrical Engineering, Stanford University, Stanford, CA 94305.

[‡]Chairman, Computer & Control Sciences Department, ENSTA, Paris, France.

[¶]Department of Aeronautics and Astronautics, Stanford University, Stanford, CA 94305.

[**]Author presenting paper. Post-doctoral Research Associate, Institute for Systems Research and Adjunct Assistant Professor of Electrical Engineering, University of Maryland, College Park, MD 20742.

Abstract

A wide variety of problems in system and control theory can be formulated (or reformulated) as convex optimization problems involving linear matrix inequalities, that is, constraints requiring an affine combination of symmetric matrices to be positive semidefinite. For a few very special cases, there are "analytical solutions" to these problems, but in general they can be solved numerically very efficiently. We introduce the reader to Linear Matrix Inequalities or LMIs, provide a brief history of LMIs in system and control theory, and discuss a few problems from systems and control that can be solved via convex optimization over LMIs.

1 Introduction

A number of problems that arise in systems and control such as optimal matrix scaling, digital filter realization, interpolation problems that arise in system identification, and robustness analysis and state-feedback synthesis via Lyapunov functions can be reduced to a handful of standard convex and quasiconvex problems that involve linear matrix inequalities. Extremely efficient interior point algorithms have recently been developed for and tested on these standard problems; further development of algorithms for these standard problems is an area of active research.

Our objective in this paper is to give a brief survey of LMIs in system and control theory: In §2, we present a brief history of LMIs in system and control theory; in §3, we introduce a few "standard" convex and quasiconvex optimization problems involving LMIs; in §4, we present a few problems from system and control theory that may be solved via LMI techniques—scaling a matrix to minimize its condition number, stability analysis of linear differential inclusions and optimization over an affine family of transfer matrices; we make a few brief remarks about solving LMI-based optimization problems in §5, and conclude with §6.

2 A Brief History of LMIs in System and Control Theory

The history of linear matrix inequalities in the analysis of dynamical systems goes back more than 100 years, when Lyapunov published his seminal work introducing what we now call Lyapunov theory. He showed that the differential equation

$$\frac{d}{dt}x(t) = Ax(t) \tag{1}$$

is stable if and only if there exists a positive definite matrix P such that

$$A^T P + PA < 0. \tag{2}$$

The requirement $P > 0$, $A^T P + PA < 0$ is what we now call a Lyapunov inequality on P, which is a special form of a linear matrix inequality. Of course, we can solve this LMI (that is, find a suitable P) by solving a Lyapunov equation.

The next major development was in the 1940's when Lur'e, Postnikov, and others in the Soviet Union applied Lyapunov's methods to some specific practical problems in control engineering, especially, the problem of stability of a control system with a nonlinearity in the actuator [18]. Although they did not explicitly form matrix inequalities, their stability criteria in fact have the form of linear matrix inequalities. These inequalities were reduced to polynomial inequalities which were then checked "by hand" (for, needless to say, small systems).

Then, in the 1960's, Yakubovich, Popov, Kalman, and other researchers succeeded in reducing the solution of the linear matrix inequalities that arose in the problem of Lur'e to simple graphical criteria, using what we now call the Kalman-Yakubovich-Popov (KYP) lemma. This resulted in the celebrated Popov criterion, Circle criterion, Tsypkin criterion, and many variations. These criteria could be applied to higher order systems, but did not gracefully or usefully extend to systems containing more than one nonlinearity. Thus, their contribution may be viewed—in the context of the history of LMIs in control theory—as showing how to solve a certain family of linear matrix inequalities by a graphical method. We should note that the important role of LMIs in control theory was already recognized in the early 1960's, especially by Yakubovich [29].

By 1971, researchers knew several methods for solving special types of LMIs: direct (for very small systems), graphical methods, and by solving Lyapunov or Riccati equations. In Willems' 1971 paper [28] we find the following striking quote:

> The basic importance of the LMI seems to be largely unappreciated. It would be interesting to see whether or not it can be exploited in computational algorithms, for example.

Willems' suggestion that linear matrix inequalities might have some advantages in computational algorithms became closer to being realized with the following observation:

> The LMIs that arise in system and control theory can be formulated as *convex optimization problems*, and hence are amenable to computer solution.

This observation was made explicitly by several researchers: Pyatnitskii and Skorodinskii [24] Horisberger and Belanger [14], to name just a few. This observation, coupled with the development of interior point methods that apply directly to convex problems involving matrix inequalities, by Nesterov and Nemirovsky in 1988, now mean that we can reliably and quickly solve many problems in systems and control for which no "analytical solution" has been found (or is likely to be found), by reducing to them to LMI problems.

3 Optimization Problems based on Linear Matrix Inequalities

A linear matrix inequality is a matrix inequality of the form

$$F(x) \overset{\Delta}{=} F_0 + \sum_{i=1}^{m} x_i F_i > 0, \tag{3}$$

where $x \in \mathbf{R}^m$ is the variable, and $F_i = F_i^T \in \mathbf{R}^{n\times n}$, $i = 0, \ldots, m$ are given. The set $\{x \mid F(x) > 0\}$ is convex, and need not have smooth boundary. (We've used strict inequality mostly for convenience; inequalities of the form $F(x) \geq 0$ are also readily handled.)

Multiple LMIs $F_1(x) > 0, \ldots, F_n(x) > 0$ can be expressed as the single LMI

$$\mathbf{diag}(F_1(x), \ldots, F_n(x)) > 0.$$

When the matrices F_i are diagonal, the LMI $F(x) > 0$ is just a set of linear inequalities. Nonlinear (convex) inequalities are converted to LMI form using Schur complements. The basic idea is as follows: the LMI

$$\begin{bmatrix} Q(x) & S(x) \\ S(x)^T & R(x) \end{bmatrix} > 0 \tag{4}$$

where $Q(x) = Q(x)^T$, $R(x) = R(x)^T$, and $S(x)$ depend affinely on x, is equivalent to

$$R(x) > 0, \; Q(x) - S(x)R(x)^{-1}S(x)^T > 0. \tag{5}$$

In other words, the set of nonlinear inequalities (5) can be represented as the LMI (4).

The matrix norm constraint $\|Z(x)\| < 1$, where $Z(x) \in \mathbf{R}^{p\times q}$ and depends affinely on x, is represented as the LMI

$$\begin{bmatrix} I & Z(x) \\ Z(x)^T & I \end{bmatrix} > 0$$

(since $\|Z\| < 1$ is equivalent to $I - ZZ^T > 0$). Note that the case $q = 1$ reduces to a general convex quadratic inequality on x.

The constraint

$$\mathbf{Tr}\, S(x)^T P(x)^{-1} S(x) < 1, \;\; P(x) > 0,$$

where $P(x) = P(x)^T \in \mathbf{R}^{n\times n}$ and $S(x) \in \mathbf{R}^{n\times p}$ depend affinely on x, is handled by introducing a new (slack) matrix variable $X = X^T \in \mathbf{R}^{p\times p}$, and the LMI (in x and X):

$$\mathbf{Tr}\, X < 1, \;\; \begin{bmatrix} X & S(x)^T \\ S(x) & P(x) \end{bmatrix} > 0.$$

We often encounter problems in which the variables are matrices, *e.g.*,

$$A^T P + PA < 0 \tag{6}$$

where $A \in \mathbf{R}^{n\times n}$ is given and $P = P^T$ is the variable. In this case we will not write out the LMI explicitly in the form $F(x) > 0$, but instead make clear which matrices are the variables. Leaving LMIs in a condensed form such as (6), in addition to saving notation, leaves open the possibility of more efficient computation.

3.1 LMI feasibility problems

Given an LMI $F(x) > 0$, the corresponding LMI Problem (LMIP) is to find x^{feas} such that $F(x^{\text{feas}}) > 0$ or determine that the LMI is infeasible. (By duality, this means: find a nonzero $G \geq 0$ such that $\mathbf{Tr}\, GF_i = 0$ for $i = 1, \ldots, m$ and $\mathbf{Tr}\, GF_0 \leq 0$.) Of course, this is a convex feasibility problem. We will say "solving the LMI $F(x) > 0$" to mean solving the corresponding LMIP.

3.2 Eigenvalue problems

The eigenvalue problem (EVP) is to minimize the maximum eigenvalue of a matrix, subject to an LMI:

$$\begin{array}{ll} \text{minimize} & \lambda \\ \text{subject to} & \lambda I - A(x) > 0, \quad B(x) > 0. \end{array}$$

Here, A and B are symmetric matrices that depend affinely on the optimization variable x. This is a convex optimization problem.

3.3 Generalized eigenvalue problems

The generalized eigenvalue problem (GEVP) is to minimize the maximum generalized eigenvalue of a pair of matrices that depend affinely on a variable, subject to an LMI constraint. The general form of a GEVP is:

$$\begin{array}{ll} \text{minimize} & \lambda \\ \text{subject to} & \lambda B(x) - A(x) > 0 \\ & B(x) > 0 \\ & C(x) > 0 \end{array}$$

where A, B and C are affine functions of x. This is a quasiconvex problem.

Note that when the matrices are all diagonal, this problem reduces to the general linear fractional programming problem. Many nonlinear quasiconvex functions can be represented in the form of a GEVP with appropriate A, B, and C (see [5]).

4 LMI Problems in Systems and Control

4.1 Minimizing condition number by scaling

The problem of improving the condition number of a matrix via diagonal scaling arises in numerical analysis and control [2, 9, 27]. (The condition number of a matrix M, denoted by $\kappa(M)$, is the ratio of the largest and smallest singular values if M is nonsingular, and ∞ otherwise.)

Given a matrix $M \in \mathbf{R}^{p\times q}$, we consider the problem:

$$\begin{array}{ll} \text{minimize} & \kappa(LMR) \\ \text{subject to} & L \in \mathbf{R}^{p\times p}, \text{ diagonal and nonsingular} \\ & R \in \mathbf{R}^{q\times q}, \text{ diagonal and nonsingular} \end{array} \qquad (7)$$

where L and R are the optimization variables. We will show that this problem can be transformed into a GEVP.

Let us assume without loss of generality that $p \geq q$ and M is full-rank.

Let us fix $\gamma > 1$. There exist nonsingular, diagonal $L \in \mathbf{R}^{p\times p}$ and $R \in \mathbf{R}^{q\times q}$ such that $\kappa(LMR) < \gamma$ if and only if

$$\mu I \leq (LMR)^*(LMR) \leq \mu\gamma^2 I$$

for some $\mu > 0$. (M^* stands for the transpose conjugate of M.)

Since we may absorb the factor $1/\sqrt{\mu}$ into L, this is equivalent to the existence of nonsingular, diagonal $L \in \mathbf{R}^{p\times p}$ and $R \in \mathbf{R}^{q\times q}$ such that

$$I \leq (LMR)^*(LMR) \leq \gamma^2 I,$$

which is the same as

$$(RR^*)^{-1} \leq M^*(L^*L)M \leq \gamma^2(RR^*)^{-1}. \tag{8}$$

This is equivalent to the existence of diagonal $P \in \mathbf{R}^{p\times p}$, $Q \in \mathbf{R}^{q\times q}$ with $P > 0$, $Q > 0$ and

$$Q < M^*PM < \gamma^2 Q. \tag{9}$$

To see this, first suppose that $L \in \mathbf{R}^{p\times p}$ and $R \in \mathbf{R}^{q\times q}$ are nonsingular and diagonal, and (8) holds. Then (9) holds with $P = L^*L$ and $Q = (RR^*)^{-1}$. Conversely, suppose that (9) holds for diagonal $P \in \mathbf{R}^{p\times p}$ and $Q \in \mathbf{R}^{q\times q}$ with $P > 0$, $Q > 0$. Then, (8) holds for $L = P^{1/2}$ and $R = Q^{-1/2}$.

Hence we can solve (7) by solving the GEVP:

$$\begin{array}{ll} \text{minimize} & \gamma^2 \\ \\ \text{subject to} & P \in \mathbf{R}^{p\times p} \text{ and diagonal} \\ & Q \in \mathbf{R}^{q\times q} \text{ and diagonal} \\ & 0 < Q < M^*PM < \gamma^2 Q \end{array}$$

We finally note that it is possible to extend the above results to the case of block-diagonal scalings.

4.2 Lyapunov function search

Consider the differential inclusion (DI)

$$\frac{dx}{dt} = A(t)x(t), \;\; A(t) \in \mathbf{Co}\left\{A_1, \ldots, A_L\right\}, \tag{10}$$

where **Co** denotes the convex hull. We ask whether the DI is stable, *i.e.*, whether all trajectories of the system (10) converge to zero as $t \longrightarrow \infty$. A sufficient condition for this is the existence of a quadratic positive function $V(z) = z^TPz$ such that $dV(x(t))/dt < 0$ for any trajectory of (10). Since

$$\frac{d}{dt}V(x(t)) = x(t)^T\left(A(t)^TP + PA(t)\right)x(t),$$

a sufficient condition for stability is the existence of $P > 0$ such that

$$A(t)^TP + PA(t) < 0, \;\; A(t) \in \mathbf{Co}\left\{A_1, \ldots, A_L\right\}. \tag{11}$$

If there exists such a P, we say the DI (10) is *quadratically stable.*

Condition (11) is equivalent to

$$P > 0, \quad A_i^T P + P A_i < 0, \quad i = 1, \ldots, L,$$

which is a linear matrix inequality in P (see for example [7, 14, 15, 25]). Thus, determining quadratic stability is an LMIP.

V is sometimes called a *simultaneous quadratic Lyapunov function* since it proves stability of each of $A_1, \ldots, A_L$.

4.3 Lyapunov functions and state feedback

Consider the system (10) with state feedback:

$$\frac{dx}{dt} = A(t)x(t) + B(t)u(t), \quad u(t) = Kx(t), \tag{12}$$

where

$$[A(t)\ B(t)] \in \mathbf{Co}\left\{[A_1\ B_1], \ldots, [A_L\ B_L]\right\}.$$

Our objective is to design the matrix K such that such that (12) is quadratically stable. This is the "quadratic stabilizability" problem (see [13], and [23, 12, 10]; related references are [30, 17, 16] and [22]).

System (12) is quadratically stable for some state state-feedback K if there exist $P > 0$ and K such that

$$(A_i + B_i K)^T P + P(A_i + B_i K) < 0, \quad i = 1, \ldots, L.$$

Note that this matrix inequality is *not* convex in P and K. However, with the linear fractional transformation $Y \triangleq P^{-1}$, $W \triangleq KP^{-1}$, we may rewrite it as

$$(A_i + B_i W Y^{-1})^T Y^{-1} + Y^{-1}(A_i + B_i W Y^{-1}) < 0.$$

Multiplying this inequality on the left and right by Y (such a congruence preserves the inequality) we get an LMI in Y and W:

$$Y A_i^T + W^T B_i^T + A_i Y + B_i W < 0, \quad i = 1, \ldots, L.$$

If this LMIP in Y and W has a solution, then the Lyapunov function $V(z) = z^T Y^{-1} z$ proves the quadratic stability of the closed-loop system with state-feedback $u(t) = WY^{-1}x(t)$.

In other words, we can synthesize a linear state feedback for the DI (10) by solving a set of simultaneous Lyapunov inequalities.

Let us also note that by synthesizing a state feedback for the DI (10), we have also synthesized a suitable state feedback for the nonlinear, time-varying, uncertain system

$$\frac{dx}{dt} = f(x, u, t)$$

if, for each x, u and t, there exists $G(x, u, t) \in \mathbf{Co}\left\{[A_1\ B_1], \ldots, [A_L\ B_L]\right\}$ so that

$$f(x, u, t) = G(x, u, t) \begin{bmatrix} x \\ u \end{bmatrix}.$$

4.4 Optimization over an affine family of linear systems

We consider an affine family of stable linear systems of the form

$$\begin{aligned} \frac{d}{dt}x(t) &= Ax(t) + B_w w(t), \\ z(t) &= C_z(\theta)x(t) + D_{zw}(\theta)w(t), \end{aligned} \tag{13}$$

where C_z and D_{zw} depend affinely on a parameter vector θ (the "design" parameter), and we assume that (A, B_w) is controllable. In this section, we show that we may choose θ such that system (13) has certain desirable properties by solving LMIPs which have θ as one of the optimization variables. The results of this section may be applied, for instance, to the problem of Finite Impulse Response filter design.

4.4.1 $\mathbf{H}_2$ norm

The $\mathbf{H}_2$ norm of system (13), if $D_{zw}(\theta) = 0$, is defined to be $\mathbf{Tr}\, C_z(\theta)W_c C_z(\theta)^T$, where $W_c \geq 0$ is the controllability Gramian which satisfies

$$AW_c + W_c A^T + B_w B_w^T = 0. \tag{14}$$

For nonzero $D_{zw}(\theta)$, the $\mathbf{H}_2$ norm is defined to be ∞. Therefore, the $\mathbf{H}_2$ norm of system (13) is less than or equal to γ if and only if the following LMI in θ and γ^2 is satisfied:

$$D_{zw}(\theta) = 0, \quad \mathbf{Tr}\, C_z(\theta)W_c C_z(\theta)^T \leq \gamma^2.$$

4.4.2 $\mathbf{H}_\infty$ norm

The $\mathbf{H}_\infty$ norm of system (13) is defined to be

$$\max_{\|w\|_2 \leq 1} \|z\|_2,$$

with $x(0) = 0$. From an extension of the Kalman-Yakubovich-Popov lemma, due to Willems [28], the constraint that the $\mathbf{H}_\infty$ norm of system (13) does not exceed γ is equivalent to an LMI in θ, P and γ^2:

$$P \geq 0, \quad \begin{bmatrix} A^T P + PA & PB_w \\ B_w^T P & -\gamma^2 I \end{bmatrix} + \begin{bmatrix} C_z(\theta)^T \\ D_{zw}(\theta)^T \end{bmatrix} \begin{bmatrix} C_z(\theta) & D_{zw}(\theta) \end{bmatrix} \leq 0.$$

4.4.3 Minimum dissipation

Suppose that w and z are vectors with the same number of components. We define the *minimum dissipation* η of as system (13) as the largest η such that

$$\int_0^T w(t)^T z(t) - \eta w(t)^T w(t)\, dt \geq 0$$

holds for all $T \geq 0$, with $x(0) = 0$. Thus passivity corresponds to nonnegative minimum dissipation.

Once again, from Willems [28], the constraint that the minimum dissipation of system (13) be at least η is equivalent to

$$P \geq 0, \quad \begin{bmatrix} A^T P + PA & PB_w - C_z(\theta)^T \\ B_w^T P - C_z(\theta) & 2\eta I - D_{zw}(\theta) - D_{zw}(\theta)^T \end{bmatrix} \leq 0,$$

an LMI in P, θ and η.

5 Solving LMI-based Problems

The most important point is:

LMIPs, EVPs and GEVPs are tractable

in a sense that can be made precise from a number of theoretical and practical viewpoints. (This is to be contrasted with much less tractable problems, *e.g.*, the general problem of robustness analysis for a system with real parameter perturbations.)

From a theoretical standpoint:

- We can immediately write down necessary and sufficient optimality conditions.
- There is a well-developed duality theory (for GEVPs, in a limited sense).
- These problems can be solved in polynomial time (indeed with a variety of interpretations of the term "polynomial-time").

The most important practical implication is that there are effective and powerful algorithms for the solution of these problems, that is, algorithms that rapidly compute the global optimum, with non-heuristic stopping criteria. Thus, on exit, the algorithms can prove that the global optimum has been obtained to within some prespecified accuracy.

There are a number of general algorithms for the solution of these problems, for example, the ellipsoid algorithm (see *e.g.*, [4, 3]). The ellipsoid method has polynomial-time complexity, and works in practice for smaller problems, but can be slow for larger problems. Other algorithms specifically for LMI-based problems are discussed in, *e.g.*, [8, 21].

Recently, various researchers [19, 5, 26, 1] have developed interior point methods for solving LMI-based problems, based on the work of Nesterov and Nemirovsky [20]. Numerical experience shows that these algorithms solve LMI problems with great efficiency. In some specific cases (see for example, [26]), these methods can solve LMI-based problems with computational effort that is comparable to that required to "evaluate" the "analytic" solutions of similar problems.

6 Conclusion

We have shown that many problems in systems and control can be cast as convex optimization problems involving LMIs. These problems do not have "analytic solutions" but can be solved extremely efficiently. The list of problems we have presented is by no means exhaustive. Other problems include:

- synthesis of gain-scheduled state-feedback
- linear controller design via Q-parametrization
- multi-criterion LQG
- interpolation problems involving scaling
- synthesis of Lyapunov functions with Popov terms for nonlinearities
- synthesis of multipliers for analysis of systems with unknown constant parameters
- analysis and design for randomly varying systems

- synthesis of quadratic Lyapunov functionals for delay systems
- problems in robust identification

We refer the reader to the forthcoming monograph [6] for details.

References

[1] F. Alizadeh. Optimization over the positive semi-definite cone : Interior–point methods and combinatorial applications. In P. M. Pardalos, editor, *Advances in Optimization and Parallel Computing*, pages 1–25. North Holland, Amsterdam, The Netherlands, 1992.

[2] F. L. Bauer. Optimally scaled matrices. *Numerische Mathematik*, 5:73–87, 1963.

[3] R. G. Bland, D. Goldfarb, and M. J. Todd. The ellipsoid method: A survey. *Operations Research*, 29(6):1039–1091, 1981.

[4] S. Boyd and C. Barratt. *Linear Controller Design: Limits of Performance*. Prentice-Hall, 1991.

[5] S. Boyd and L. El Ghaoui. Method of centers for minimizing generalized eigenvalues. *Linear Algebra and Applications, special issue on Numerical Linear Algebra Methods in Control, Signals and Systems*, 188:63–111, July 1993.

[6] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. Linear matrix inequalities in systems and control theory, 1993. Monograph in preparation.

[7] S. Boyd and Q. Yang. Structured and simultaneous Lyapunov functions for system stability problems. *Int. J. Control*, 49(6):2215–2240, 1989.

[8] J. Cullum, W. Donath, and P. Wolfe. The minimization of certain nondifferentiable sums of eigenvalues of symmetric matrices. *Math. Programming Study*, 3:35–55, 1975.

[9] G. Golub and J. Varah. On a characterization of the best ℓ_2 scaling of a matrix. *SIAM J. on Numerical Analysis*, 11(3):472–479, June 1974.

[10] K. Gu. Linear feedback stabilization of arbitrary time-varying uncertain systems. In *Proc. American Control Conf.*, Chicago, June 1992.

[11] W. M. Haddad, J. P. How, S. R. Hall, and D. S. Bernstein. Extensions of mixed-μ bounds to monotonic and odd monotonic nonlinearities using absolute stability theory. In *Proc. IEEE Conf. on Decision and Control*, Tucson, Arizona, December 1992.

[12] C. V. Hollot. Bound invariant Lyapunov functions: a means for enlarging the class of stabilizable uncertain systems. *Int. J. Control*, 46(1):161–184, 1987.

[13] C. V. Hollot and B. R. Barmish. Optimal quadratic stabilizability of uncertain linear systems. *Proc. 18th Allerton Conf. on Communication, Control and Computing, University of Illinois, Monticello*, pages 697–706, 1980.

[14] H. P. Horisberger and P. R. Belanger. Regulators for linear, time invariant plants with uncertain parameters. *IEEE Trans. Aut. Control*, AC-21:705–708, 1976.

[15] V. A. Kamenetskii. Absolute stability and absolute instability of control systems with several nonlinear nonstationary elements. *Automation and Remote Control*, 44(12):1543–1552, 1983.

[16] P. P. Khargonekar, I. R. Petersen, and K. Zhou. Robust stabilization of uncertain linear systems: quadratic stabilizability and $\mathbf{H}_\infty$ control theory. *IEEE Trans. Aut. Control*, 35(3):356–361, March 1990.

[17] P. P. Khargonekar and M. A. Rotea. Stabilization of uncertain systems with norm bounded uncertainty using control Lyapunov functions. *Proc. IEEE Conf. on Decision and Control*, 1:503–507, 1988.

[18] A. I. Lur'e. *Some Nonlinear Problems in the Theory of Automatic Control.* H. M. Stationery Off., London, 1957. In Russian, 1951.

[19] Yu. Nesterov and A. Nemirovsky. An interior point method for generalized linear-fractional programming. *submitted to Math. Programming, Series B*, 1991.

[20] Yu. Nesterov and A. Nemirovsky. *Interior point polynomial methods in convex programming: Theory and applications.* SIAM, 1993.

[21] M. Overton. On minimizing the maximum eigenvalue of a symmetric matrix. *SIAM J. on Matrix Analysis and Applications*, 9(2):256–268, 1988.

[22] A. Packard and J. C. Doyle. Quadratic stability with real and complex perturbations. *IEEE Trans. Aut. Control*, 35(2):198–201, February 1990.

[23] I. R. Petersen. Quadratic stabilizability of uncertain linear systems: Existence of a nonlinear stabilizing control does not imply existence of a stabilizing control. *IEEE Trans. Aut. Control*, AC-30(3):291–293, 1985.

[24] E. S. Pyatnitskii and V. I. Skorodinskii. Numerical methods of Lyapunov function construction and their application to the absolute stability problem. *Syst. Control Letters*, 2(2):130–135, August 1982.

[25] E. S. Pyatnitskii and V. I. Skorodinskii. Numerical method of construction of Lyapunov functions and absolute stability criteria in the form of numerical procedures. *Automation and Remote Control*, 44(11):1427–1437, 1983.

[26] L. Vandenberghe and S. Boyd. Primal-dual potential reduction method for problems involving matrix inequalities. To be published in *Math. Programming*, 1993.

[27] G. A. Watson. An algorithm for optimal ℓ_2 scaling of matrices. *IMA Journal of Numerical Analysis*, 11:481–492, 1991.

[28] J. C. Willems. Least squares stationary optimal control and the algebraic Riccati equation. *IEEE Trans. Aut. Control*, AC-16(6), December 1971.

[29] V. A. Yakubovich. The method of matrix inequalities in the stability theory of nonlinear control systems, *I, II, III. Automation and Remote Control*, 25-26(4):905–917, 577–592, 753–763, April 1967.

[30] K. Zhou and P. P. Khargonekar. Robust stabilization of linear systems with norm-bounded time-varying uncertainty. *Syst. Control Letters*, 10:17–20, 1988.

Computing the Poles of Autoregressive Models from the Reflection Coefficients

Gregory S. Ammar[1] Daniela Calvetti[2] Lothar Reichel[3]

Abstract

A new approach to the computation of the poles of a stable autoregressive system from the reflection coefficients is proposed. Equivalently, we compute the zeros of Szegő polynomials from the associated Schur parameters. The numerical method utilizes an efficient algorithm for computing the (unimodular) zeros of a unitary Hessenberg matrix; this step can be regarded as the computation of the poles of an associated lossless system. These eigenvalues are then used as starting points for a continuation procedure for finding the zeros of the desired polynomial. The procedure is efficient and parallelizable, and may therefore be suitable for real-time applications.

1. Introduction

Autoregressive models are of fundamental importance in time series analysis and discrete-time control. For example, in linear prediction, one is often given the autocorrelation matrix of a wide sense stationary real-valued signal $\{x_j\}_{j=-\infty}^{\infty}$. This matrix is a real symmetric positive definite Toeplitz matrix

$$M_{n+1} = [\mu_{j-k}]_{j,k=0}^{n}.$$

For each positive integer $p \leq n$, the time series can be modeled as the response of an autoregressive (AR) process of order p driven by white noise; i.e.,

$$x_j = w_j - \sum_{k=1}^{p} \alpha_k x_{j-k}, \tag{1}$$

where $\{w_j\}_{j=-\infty}^{\infty}$ is a white noise process. The original time series is approximated by the AR(p) process $\{\hat{x}_j\}_{j=-\infty}^{\infty}$, given by

$$\hat{x}_j = -\sum_{k=1}^{p} \alpha_k x_{j-k}.$$

The coefficients $\{\alpha_k\}_{k=1}^{p}$ that minimize the mean square prediction error are determined by solving the *Yule-Walker equations* of order p,

$$M_p \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_p \end{bmatrix} = - \begin{bmatrix} \mu_1 \\ \vdots \\ \mu_{p+1} \end{bmatrix}. \tag{2}$$

---

[1] Department of Mathematical Sciences, Northern Illinois University, DeKalb, IL 60115. E-mail: ammar@math.niu.edu

[2] Department of Pure and Applied Mathematics, Stevens Institute of Technology, Hoboken, NJ 07030. E-mail: calvetti@mathrs1.stevens-tech.edu.

[3] Department of Mathematics and Computer Science, Kent State University, Kent, OH 44242. E-Mail: reichel@mcs.kent.edu. Research supported in part by the National Science Foundation under grants DMS-9002884 and DMS-9205531.

In the z-transform domain, the filter (1) takes the form

$$X(z) = \frac{1}{A_p(z)} W(z),$$

where

$$A_p(z) = \sum_{k=0}^{p} \alpha_k z^{-k} \qquad (\alpha_0 \equiv 1).$$

Since the transfer function $(A_p(z))^{-1}$ has no (finite) zeros, AR processes are also referred to as *all-pole filters*. The above assumptions guarantee that the process (1) is *stable*; that is, the zeros of $A_p(z)$ are all of modulus less than one. See, for example, [17, Chapter 6].

The entries of the Toeplitz matrix M_{n+1} can be regarded as the moments of a distribution function on the unit circle in the complex plane. These moments can be used to construct the associated inner product $< \cdot, \cdot >$ on polynomials of degree at most n by the linear extension of the identity $< \lambda^j, \lambda^k > := \mu_{j-k}$. The monic polynomials $\psi_j(\lambda)$ that are orthogonal with respect to this measure are known as Szegő polynomials in this context. They are also called Levinson polynomials and backward predictor polynomials in the engineering literature.

Szegő polynomials can be recursively computed using the well-known Levinson-Durbin algorithm. This algorithm is equivalent with the use of the recursions formulas introduced by Szegő (see, e.g., [21, 15]) to construct the coefficients of the Szegő polynomials in the power basis. In particular, the Szegő recursions for constructing the monic polynomials ψ_j are given by

$$\begin{aligned} \psi_{j+1}(\lambda) &= \lambda\psi_j(\lambda) + \gamma_{j+1}\tilde{\psi}_j(\lambda), \\ \tilde{\psi}_{j+1}(\lambda) &= \lambda\gamma_{j+1}\psi_j(\lambda) + \tilde{\psi}_j(\lambda), \end{aligned}$$

where $\psi_0(\lambda) = \tilde{\psi}_0(\lambda) = 1$, and

$$\gamma_{j+1} = - < 1, \lambda\psi_j > / < \psi_j, \psi_j > .$$

The recursion coefficients γ_j are known as *Schur parameters*, and also as *Szegő parameters, reflection coefficients*, and *partial correlation coefficients.*

It follows from the Szegő recursions that the auxiliary polynomials $\tilde{\psi}_j(\lambda)$ satisfy $\tilde{\psi}_j(\lambda) = \lambda^j\psi_j(\lambda^{-1})$. These are known as *forward predictor polynomials* in linear prediction. In fact, $\tilde{\psi}_p(\lambda) = A_p(\lambda^{-1})$ for $1 \leq p \leq n$. Consequently, the zeros of $A_p(\lambda)$ and $\psi_p(\lambda)$ coincide.

The coefficients of the Szegő polynomial ψ_n in the power basis provide the *recursive realization* of the autoregressive model of order n. On the other hand, the Schur parameters provide the realization of the AR model as a lattice filter. The latter realization has several advantages over the former. For example, the filter is stable if and only if each γ_j has modulus less than one. This condition can be easily enforced, even if the parameters are corrupted by noise. In contrast, it is not clear how one can enforce the stability of a filter in recursive form.

The zeros of the Szegő polynomial ψ_n provide another parameterization of the AR model. Moreover, the knowledge of the poles is often desired. However, the difficulties in reliably and efficiently computing of the zeros of the polynomial ψ_n often discourages their use in practice.

The most straightforward approach to computing the zeros of the polynomials ψ_n is to use a polynomial root-finder; see, e.g., [20, Sections 5.5-5.10] for a description of a few methods. For several of these methods it is important that the zeros be determined in an appropriate order; otherwise severe loss of accuracy in the computed roots may result. Moreover, in order to obtain globally and quickly converging methods, several iterative techniques may have to be employed. These considerations can make software for reliable and quickly converging rootfinders fairly complicated. Consequently, the problem of finding zeros of a polynomial should be replaced by the problem of computing eigenvalues of a matrix when possible, see, e.g., the the discussion in [20, Section 5.5].

The most obvious way to replace the rootfinding problem with an eigenvalue problem is to consider the Frobenius matrix (companion matrix) associated with the power basis coefficients of the polynomial ψ_n. The Schur parameters determine another matrix whose eigenvalues are the roots of ψ_n. This is the upper Hessenberg matrix

$$H_n = H(\gamma_1, \dots, \gamma_n) = G_1(\gamma_1)G_2(\gamma_2)...G_{n-1}(\gamma_{n-1})\tilde{G}_n(\gamma_n), \tag{3}$$

where $G_j(\gamma_j)$ is the (orthogonal) Givens matrix of order n that operates in the $(j, j+1)$ coordinate plane,

$$G_j(\gamma_j) = \begin{bmatrix} I_{j-1} & & & \\ & -\gamma_j & \sigma_j & \\ & \sigma_j & \gamma_j & \\ & & & I_{n-j-1} \end{bmatrix}, \qquad 1 \le j < n.$$

Here I_k denotes the $k \times k$ identity matrix. The matrix $\tilde{G}_n(\gamma_n)$ is the truncated Givens matrix

$$\tilde{G}_n(\gamma_n) = \begin{bmatrix} I_{n-1} & \\ & -\gamma_n \end{bmatrix}. \tag{4}$$

The connection between Szegő polynomials and Hessenberg matrices of the form (3) was uncovered by Gragg [12], who showed that the eigenvalues of H_n are the zeros of ψ_n. More precisely,

$$\psi_n(z) = \det[zI_n - H_n]. \tag{5}$$

If we use the QR-algorithm, see [10, Chapter 7] or [6], for computing the eigenvalues of H_n, then $O(n^3)$ arithmetic operations are required. It is the purpose of this paper to outline a new algorithm that requires only $O(n^2)$ arithmetic operations, and, moreover, has a structure that lends itself well to parallel computation. Our method is based on the observation that the matrix H_n is a rank-one modification of an orthogonal upper Hessenberg matrix.

2. Orthogonal and Unitary Hessenberg Eigenproblems

Orthogonal upper Hessenberg matrices form an important special case of matrices in the form (3).

Proposition 1 *The following statements are equivalent:*

- *The autocorrelation matrix M_{n+1} is nonnegative definite and of rank n.*
- *The Schur parameters satisfy $|\gamma_j| < 1$ for $1 \leq j < n$, and $|\gamma_n| = 1$.*
- *The zeros of $\psi_n(\lambda)$ are pairwise distinct and of unit modulus.*
- *H_n is an orthogonal Hessenberg matrix.*

The eigenproblem for orthogonal, and more generally, unitary Hessenberg matrices, is particularly nice. These are normal matrices with distinct eigenvalues, so numerical complications due to multiple eigenvalues and nontrivial Jordan structure is avoided. Moreover, several efficient ($O(n^2)$) algorithms have been presented for solving eigenproblems for orthogonal and unitary Hessenberg matrices. These include the unitary Hessenberg QR algorithm [11], an algorithm for the orthogonal Hessenberg eigenproblem using two half-size SVDs [3], divide-and-conquer methods [13, 14, 5], an approach based on an associated generalized eigenproblem [7], and a bisection method [8].

Eigenproblems for orthogonal and unitary Hessenberg matrices are useful in frequency estimation procedures that are based on finding the poles of lossless AR models such as Pisarenko's method and the Composite Sinusoidal Modeling method [4, 19].

In contrast, the eigenproblem for H_n when $|\gamma_n| < 1$ has a less tractable structure. The eigenvalues are of modulus less than one, but they can be of arbitrary algebraic multiplicity (although the geometric multiplicity of each eigenvalue is equal to one). Moreover, the algorithms for unitary Hessenberg matrices do not extend to the "subunitary" matrices H_n. For example, the QR algorithm does not readily extend because an orthogonal similarity transformation of H_n results in a Hessenberg matrix that cannot, in general, be represented in the form (3).

We now propose the following approach to efficiently compute the eigenvalues of $H_n = H(\gamma_1, \ldots, \gamma_n)$ with each $|\gamma_j| < 1$. Define the function

$$f(\lambda, t) = \det[\lambda I - H_n(t)], \tag{6}$$

where

$$H_n(t) = H(\gamma_1, \ldots, \gamma_{n-1}, \alpha(t))$$

and

$$\alpha(t) = (1-t)\beta + t\gamma_n$$

with $|\beta| = 1$. Then the zeros of $f(\lambda, 0)$ are the eigenvalues of an orthogonal Hessenberg matrix, which can be quickly computed. These eigenvalues are then used as starting points for a continuation method, in which we track the zeros of $f(\lambda, t)$ as t varies from 0 to 1. The endpoints of these paths are then the eigenvalues of H_n.

The parameter β is chosen so that $H_n(0)$ is the closest orthogonal Hessenberg matrix to $H_n(1)$ in any unitarily invariant norm. In particular, we choose [19]

$$\beta = \begin{cases} \gamma_n/|\gamma_n| & \text{if } \gamma_n \neq 0, \\ 1 & \text{if } \gamma_n = 0 \end{cases}.$$

3. Predictor-Corrector Methods for Path Following

Our approach to the design of continuation methods for following the eigenvalues paths is based on the algorithm presented by Li and Zeng [18] and on ideas given by Allgower and Georg [1, 2].

Having determined the eigenvalues $\lambda_j(0)$, $1 \leq j \leq n$, of $H_n(0)$, we apply a continuation method to track the eigenvalues of $H_n(t)$ as t increases in order to determine the spectrum of $H_n(1)$. This method is based on the identity

$$\det[\lambda_j(t)I_n - H_n(t)] = 0, \qquad 0 \leq t \leq 1, \qquad 1 \leq j \leq n, \tag{7}$$

which, in view of (6), yields

$$0 = \frac{d}{dt} f(\lambda_j(t), t) = f_\lambda(\lambda_j(t), t)\frac{d\lambda_j(t)}{dt} + f_t(\lambda_j(t), t) \tag{8}$$

for $1 \leq j \leq n$.

Thus, if $f(\lambda, t)$ has only a simple zero at $\lambda_j(t)$, we obtain the differential equation

$$\frac{d}{dt}\lambda_j(t) = \frac{-f_t(\lambda_j(t), t)}{f_\lambda(\lambda_j(t), t)}, \qquad 1 \leq j \leq n. \tag{9}$$

Moreover, $f(\lambda, t)$, as well as f_λ and f_t, can be efficiently evaluated in $O(n)$ arithmetic operations using the Szegő recursions.

The eigenvalues $\lambda_j(t)$, $1 \leq j \leq n$, of $H_n(t)$ are continuous functions of t. Moreover, each $\lambda_j(t)$ is a piecewise analytic function of t, for $0 \leq t \leq 1$, whose only singularities are algebraic; see [16, Chapter 2]. A discussion on properties of the map $t \rightarrow \lambda_j(t)$ can also be found in [18]. The purpose of the continuation method is to track the eigenvalue paths

$$\{\lambda_j(t),\ 0 \leq t \leq 1\}, \qquad 1 \leq j \leq n,$$

in order to determine the eigenvalues $\lambda_j(1)$ of $H_n = H_n(1)$ from the eigenvalues $\lambda_j(0)$ of $H_n(0)$. Toward this end, we apply a scheme described by Deuflhard and Hohmann [9, Section 4.4.2] that uses Euler's method as a predictor and Newton's method as a corrector. This scheme is closely related to the methods described in [1, Chapter 6].

Assume for the moment that $\lambda_j(t)$ is a simple eigenvalue of $H_n(t)$ for $0 \leq t \leq 1$, and let $\lambda_j(t_k)$ be explicitly known for a parameter value t_k such that $0 \leq t_k < 1$. Let h be a positive stepsize such that $t_{k+1} = t_k + h \leq 1$. An approximation $\lambda_j^{(1)}$ of $\lambda_j(t_{k+1})$ is determined by applying Euler's method with constant step length h to the differential equation (9). We obtain

$$\lambda_j^{(1)} := \lambda_j(t_k) + h\frac{-f_t(\lambda_j(t_k), t_k)}{f_\lambda(\lambda_j(t_k), t_k)}. \tag{10}$$

Typically, $\lambda_j^{(1)}$ is not a sufficiently accurate approximation of $\lambda_j(t_{k+1})$. We determine improved approximations $\lambda_j^{(2)}$, $\lambda_j^{(3)}, \ldots$ of $\lambda_j(t_{k+1})$ by applying Newton's method for the equation $f(z, t_{k+1}) = 0$ and start the iterations with $z = \lambda_j^{(1)}$. We obtain

```
for $\ell := 1, 2, 3, \ldots$ until convergence do
    $h^{(\ell)} := f(\lambda_j^{(\ell)}, t_{k+1}) / f_\lambda(\lambda_j^{(\ell)}, t_{k+1})$;
    $\lambda_j^{(\ell+1)} := \lambda_j^{(\ell)} - h^{(\ell)}$;
$\lambda_j(t_{k+1}) := \lambda_j^{(\ell+1)}$.
```

Following Deuflhard and Hohmann [9], we require that the Newton corrections $h^{(\ell)}$ satisfy

$$\left|\frac{h^{(\ell+1)}}{h^{(\ell)}}\right| \leq \frac{1}{2} \tag{11}$$

for all $\ell \leq 1$. If the condition (11) is violated for some value of ℓ, then we reduce the step length h according to

$$h := 2^{-1/2}h$$

and determine a new value $\lambda_j^{(1)}$ by Euler's method (10) with the reduced step length. The value $\lambda_j^{(1)}$ now approximates $\lambda_j(t_{k+1})$ with $t_{k+1} := t_k + h$. More accurate approximations $\lambda_j^{(2)}, \lambda_j^{(3)}, \ldots$ of $\lambda_j(t_{k+1})$ are, as above, determined by Newton's method applied to the equation $f(z, t_{k+1}) = 0$.

Conversely, assume that (11) is satisfied for all Newton corrections computed, and that moreover

$$\left|\frac{h^{(2)}}{h^{(1)}}\right| \leq \frac{1}{8}.$$

Then, after determining $\lambda(t_{k+1})$, we increase, if possible, the step length h by a factor $2^{1/2}$ according to

$$h := \min\{2^{1/2}h,\ 1 - t_{k+1}\}$$

before defining $t_{k+2} := t_{k+1} + h$.

Assuming that the number of arithmetic operations required to determine $\lambda_j(1)$ from $\lambda_j(0)$ for each j grows only linearly with n, the spectrum of H_n can be computed in only $O(n^2)$ arithmetic operations. Also note that, in principal, each eigenvalue $\lambda_j(1)$ can be determined from $\lambda_j(0)$, independently of the other eigenvalues. This makes the method well suited for parallel computation.

For some matrices $H_n(1)$, however, tracking the eigenvalue paths can require that certain eigenvalues $\lambda_j(t)$ be determined for a larger number of increasing values of t. This may occur when an eigenvalue path has large curvature or when there is a bifurcation of eigenvalue paths.

Generically, only two types of bifurcations can occur. The first type occurs when a conjugate pair of eigenvalue paths meet on the real axis and become two real paths moving on opposite directions. The second type occurs in an opposite situation, when two real eigenvalues meet and become a complex conjugate pair. Other bifurcations are quite unlikely, although possible. However, we have not yet observed the bifurcation of two non-conjugate complex paths. Consequently, we are developing our algorithm to handle bifurcations on the real axis only, using the approach of Li and Zeng [18].

After candidates for the terminal eigenvalues $\{\lambda_j(1)\}_{j=1}^n$ are obtained, we check these for duplication. If there is duplication, we check the derivatives to see if the duplicated eigenvalue is a multiple eigenvalue. If it is not, we repeat the path-following scheme on these two paths with more stringent conditions (e.g., with a smaller initial stepsize h_0). We also try again for those paths that did not terminate properly.

Finally, if the modified h_0 is too small, the procedure terminates with an error flag to signal to the user that all the eigenvalues were not found. If the user is not satisfied with the eigenvalues that were found, a backup procedure, such as the QR algorithm for general upper Hessenberg matrices, can be employed. Preliminary experiments indicate the procedure will efficiently and successfully compute the eigenvalues of many matrices H_n.

References

[1] E. Allgower and K. Georg. *Numerical Continuation Methods.* Springer-Verlag, Berlin, 1990.

[2] E. L. Allgower and K. Georg. Continuation and path following. *Acta Numerica*, pages 1–64, 1993.

[3] G. S. Ammar, W. B. Gragg, and L. Reichel. On the eigenproblem for orthogonal matrices. In *Proceedings of the 25th Conference on Decision and Control*, pages 1963–1966, 1986.

[4] G. S. Ammar, W. B. Gragg, and L. Reichel. Determination of Pisarenko frequency estimates as eigenvalues of an orthogonal matrix. In F. T. Luk, editor, *Advanced Algorithms and Architectures for Signal Processing II*, pages 143–145, 1987.

[5] G. S. Ammar, L. Reichel, and D. C. Sorensen. An implementation of a divide and conquer algorithm for the unitary eigenproblem. *ACM Trans. Math. Software*, 18:292–307, 1992.

[6] E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, S. Ostrouchov, and D. Sorensen. *LAPACK Users' Guide.* SIAM, Philadelphia, 1992.

[7] A. Bunse-Gerstner and L. Elsner. Schur parameter pencils for the solution of the unitary eigenproblem. *LAA*, 154–156:741–778, 1991.

[8] A. Bunse-Gerstner and C. He. A Sturm sequence of polynomials for unitary Hessenberg matrices. Preprint, 1992.

[9] P. Deuflhard and A. Hohmann. *Numerische Mathematik.* de Gruyter, Berlin, 1991.

[10] G. H. Golub and C. F. Van Loan. *Matrix Computations.* Johns Hopkins University Press, second edition, 1989.

[11] W. B. Gragg. The QR algorithm for unitary Hessenberg matrices. *J. Comput. Appl. Math.*, 16:1–8, 1986.

[12] W. B. Gragg. Positive definite Toeplitz matrices, the Arnoldi process for isometric operators, and Gaussian quadrature on the unit circle. *J. Comput. Appl. Math.*, 46:183–198, 1993. Originally published (in Russian) in: E.S. Nikolaev, Ed., *Numerical Methods in Linear Algebra,* Moscow Univ. Press, 1982, pp. 16–32.

[13] W. B. Gragg and L. Reichel. A divide and conquer method for the unitary eigenproblem. In M. T. Heath, editor, *Hypercube Multiprocessors 1987*, SIAM Philadelphia, 1987, pages 639–647.

[14] W. B. Gragg and L. Reichel. A divide and conquer method for unitary and orthogonal eigenproblems. *Numer. Math.*, 57:695–718, 1990.

[15] U. Grenander and G. Szegő. *Toeplitz Forms and Their Applications.* Chelsea, New York, second edition, 1984.

[16] T. Kato. *Perturbation Theory for Linear Operators.* Sprinver-Verlag, second edition, 1976.

[17] S. M. Kay. *Modern Spectral Estimation: Theory and Application.* Prentice Hall, 1988.

[18] T. Y. Li and Z. Zeng. Homotopy-determinant algorithm for solving nonsymmetric eigenvalue problems. *Math. Comp.*, 59:483–502, 1992.

[19] L. Reichel and G. S. Ammar. Fast approximation of dominant harmonics by solving an orthogonal eigenproblem. In J. C. McWhirter, editor, *Mathematics in Signal Processing II*, pages 575–591. Clarendon Press, 1990.

[20] J. Stoer and R. Bulirsch. *Introduction to Numerical Analysis.* Springer-Verlag, 1980.

[21] G. Szegő. *Orthogonal Polynomials.* American Math. Soc., Providence, 1939.

Computation of the Stable Deflating Subspace of a Symplectic Pencil using Structure Preserving Orthogonal Transformations[1]

R.V. Patel

Department of Electrical & Computer Engineering
Concordia University
Montreal, Quebec, Canada H3G 1M8

Abstract

In this paper, we consider the problem of computing a basis for the stable deflating subspace of a symplectic pencil. An algorithm for computing a "Hessenberg-type" condensed form of the pencil is first described. This algorithm uses only orthogonal structure preserving transformations. The condensed form is then used to develop a QZ-type algorithm incorporating a block implementation of multiple shifts to obtain an upper block triangular form of the symplectic pencil. A basis for the stable deflating subspace can then be obtained directly from this block triangular pencil.

1. Introduction

A matrix $S \in \mathbb{R}^{2n \times 2n}$ is called a *symplectic matrix* if $SJS^T = J$ where $J = \begin{bmatrix} 0_n & I_n \\ -I_n & 0_n \end{bmatrix}$
A pencil $K - \lambda L$, with $K,\ L \in \mathbb{R}^{2n \times 2n}$ is called a *symplectic pencil* if $KJK^T = LJL^T$

The problem of computing a basis for the deflating subspace of a symplectic pencil $K - \lambda L$ corresponding to a certain subset of its eigenvalues, *e.g.*, the set of stable eigenvalues arises [1,2] in solving the well-known discrete-time algebraic Riccati equation (DARE):

$$\mathbf{DR}(P) \equiv A^T P A - P - A^T P B (D + B^T P B)^{-1} B^T P A + H = 0 \qquad (1)$$

where $H = C^T C \in \mathbb{R}^{n \times n}$ and $D \in \mathbb{R}^{m \times m} (m \leq n)$ are positive semidefinite and positive definite matrices respectively; $A \in \mathbb{R}^{n \times n}$ and $B \in \mathbb{R}^{n \times m}$. It can be shown that if (A, B) is a stabilizable pair and (A, C) is a detectable pair, then (1) has a unique positive semidefinite solution $P = P^T \in \mathbb{R}^{n \times n}$ which is a stabilizing solution, *i.e.*, the matrix $A_c = A - B(D + B^T P B)^{-1} B^T P A$ has all its eigenvalues inside the unit circle with centre at the origin of the complex plane.

A well-known approach for solving the above problem numerically is the *Generalized Schur Vector* approach [1,2] which is based on the properties of the regular symplectic pencil $K - \lambda L$ where

$$K = \begin{bmatrix} A & 0 \\ -H & I \end{bmatrix}, \quad \text{and} \quad K = \begin{bmatrix} I & F \\ 0 & A^T \end{bmatrix} \qquad (2)$$

with $F = BD^{-1}B^T$. The algorithms in [1,2] involve the reduction of the pencil $K - \lambda L$ to a generalized real Schur form (GRSF)[2]. The eigenvalues of the resulting pencil are then re-ordered such that the n stable eigenvalues are in the top left corner. From the accumulated transformations, one can easily obtain a basis for the stable deflating subspace of the pencil $K - \lambda L$, and compute the positive semi-definite solution of (1). This algorithm does not take into account the special (symplectic) structure of the pencil $K - \lambda L$, so that the transformations employed in this algorithm destroy this structure. On the other hand, as the following result shows, it is possible to carry out the required transformations on KL such that its symplectic structure is preserved.

---

[1] This research was supported by the Natural Sciences & Engineering Research Council of Canada under Grant OGP0001345.

[2] A pencil in generalized real Schur form is a real upper quasi-triangular pencil with scalars along the diagonal for each real eigenvalue and (2×2) blocks for each pair of complex-conjugate eigenvalues.

Theorem 1.1: If the regular symplectic pencil $K - \lambda L$ has no eigenvalues of modulus 1, then there exist an orthogonal matrix $Q \in \mathbb{R}^{2n\times 2n}$ and an orthogonal symplectic matrix

$$Z = \begin{bmatrix} Z_{11} & Z_{12} \\ -Z_{12} & Z_{11} \end{bmatrix}$$

with $Z_{11}, Z_{12} \in \mathbb{R}^{n\times n}$, such that

$$Q(K - \lambda L)Z = \begin{bmatrix} T_{11} & T_{12} \\ 0_n & T_{22} \end{bmatrix} - \lambda \begin{bmatrix} R_{11} & R_{12} \\ 0_n & R_{22} \end{bmatrix} \equiv T - \lambda R \tag{3}$$

and $T - \lambda R$ is a symplectic pencil in GRSF, where $T_{11} - \lambda R_{11}$ has all its eigenvalues inside the unit circle and $T_{22} - \lambda R_{22}$ has all its eigenvalues (including infinite ones) outside the unit circle.

Theorem 1.1 and its proof in [3] show that it is possible to reduce $K - \lambda L$ using structure-preserving *orthogonal* transformations to the GRSF (3), but it provides an *efficient* QZ type algorithm for doing so only for a special case, namely when $rank(F) = 1$ or $rank(H) = 1$. In order to obtain a stabilizing solution of the DARE, we need to compute the eigenvalues of the pencil $K - \lambda L$ as well as its deflating subspace corresponding to the set of stable eigenvalues. The algorithm in [3] enables us to do this for the special case considered there. However, to the best of our knowledge, a numerically stable algorithm for the general case which preserves or uses the symplectic structure of the pencil $K - \lambda L$ has yet to be developed.

In this paper, we develop an algorithm (for the general case) which uses structure-preserving orthogonal transformations to reduce the pencil $K - \lambda L$ to a block upper-triangular form which differs from (3) only in that the pencils $T_{ii} - \lambda R_{ii}$, $i = 1,2$ are not in GRSF. However, the fact that each pencil only has either stable or unstable eigenvalues allows us to obtain a basis for the stable deflating subspace of $K - \lambda L$ from the appropriate columns of Z. In the next section, we obtain a condensed form for $K - \lambda L$ using structure preserving orthogonal transformations, and then in Section 3 we employ a QZ-type algorithm with a multiple shift strategy where n shifts are applied at a time. In order to determine the required shifts, we use a recent efficient and reliable method [4] for computing the eigenvalues of $K - \lambda L$. The algorithm proposed in this paper can be regarded as an analog, for the case of symplectic pencils, of one proposed recently by Ammar and Mehrmann [6] for Hamiltonian matrices.

2. A Condensed Form for $K - \lambda L$

There are two types of orthogonal symplectic matrices that are particularly useful in performing structure-preserving transformations on symplectic pencils. The first type consists of Householder symplectic matrices defined by

$$P(k,u) = \begin{bmatrix} \hat{P} & 0_n \\ 0_n & \hat{P} \end{bmatrix}, \qquad \hat{P} \in \mathbb{R}^{n\times n}$$

where

$$\hat{P} = I_n - \frac{2uu^T}{u^T u}$$

and

$$u^T = [0, \ \dots \ , \ 0, \ u_k, \ \dots \ , \ u_n] \neq 0^T$$

The second type consists of Givens (Jacobi) symplectic matrices defined by

$$G(k, c, s) = \begin{bmatrix} C & S \\ -S & C \end{bmatrix}, \qquad C, S \in \mathbb{R}^{n \times n}$$

where

$$C = diag[I_{k-1}, \ c \ , \ I_{n-k}]$$
$$S = diag[0_{k-1}, \ s \ , \ 0_{n-k}]$$

and $c^2 + s^2 = 1$. Algorithms H and J in [7] show how $P(k, u)$ and $G(k, c, s)$ can be determined to zero specific entries in a vector.

Next, we present an algorithm for reducing $K - \lambda L$ to a condensed form using an orthogonal matrix Q and an orthogonal symplectic matrix Z. *i.e.*,

$$K - \lambda L := Q(K - \lambda L)Z \equiv \begin{bmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{bmatrix} - \lambda \begin{bmatrix} L_{11} & L_{12} \\ 0_n & L_{22} \end{bmatrix} \tag{4}$$

where K_{21}, $L_{22} \in \mathbb{R}^{n \times n}$ are upper-triangular matrices, $L_{11} \in \mathbb{R}^{n \times n}$ is a lower triangular matrix, and $K_{22} \in \mathbb{R}^{n \times n}$ is an upper Hessenberg matrix.

To illustrate the reductions performed by the algorithm, we consider the following case ($n = 5$). For the sake of generality, we shall not assume any special structure for the submatrices of K or L as shown in (2). Our only assumption is that $K - \lambda L$ is a symplectic pencil.

$$K = \begin{bmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{bmatrix} = \left[\begin{array}{ccccc|ccccc} \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \end{array}\right]$$

$$L = \begin{bmatrix} L_{11} & L_{12} \\ L_{21} & L_{22} \end{bmatrix} = \left[\begin{array}{ccccc|ccccc} \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \end{array}\right]$$

The aim is to reduce $K - \lambda L$ to the condensed form shown in (4). We do this using the orthogonal transformations mentioned above. First, we use Householder transformations to reduce L to an upper block triangular form with $L_{21} = 0$, L_{22} an upper triangular matrix, and L_{11} a lower triangular matrix. This corresponds to transformations on the rows of $K - \lambda L$, *i.e.*,

$$K - \lambda L := Q_1(K - \lambda L)$$

This reduction can be achieved using a slight variation of the QR decomposition of L. The pencil $K - \lambda L$ will have the following structure:

$$\left[\begin{array}{ccccccccccc}
\times & \times & \times & \times & \times & & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & & \times & \times & \times & \times & \times \\[2ex]
\times & \times & \times & \times & \times & & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & & \times & \times & \times & \times & \times
\end{array}\right] - \lambda \left[\begin{array}{ccccccccccc}
\times & 0 & 0 & 0 & 0 & & \times & \times & \times & \times & \times \\
\times & \times & 0 & 0 & 0 & & \times & \times & \times & \times & \times \\
\times & \times & \times & 0 & 0 & & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & 0 & & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & & \times & \times & \times & \times & \times \\[2ex]
0 & 0 & 0 & 0 & 0 & & \times & \times & \times & \times & \times \\
0 & 0 & 0 & 0 & 0 & & 0 & \times & \times & \times & \times \\
0 & 0 & 0 & 0 & 0 & & 0 & 0 & \times & \times & \times \\
0 & 0 & 0 & 0 & 0 & & 0 & 0 & 0 & \times & \times \\
0 & 0 & 0 & 0 & 0 & & 0 & 0 & 0 & 0 & \times
\end{array}\right]$$

Next, we zero the elements $k_{10,i}$ into $k_{10,i-1}$, $i = 4, 3, 2$ using 2×2 modified Householder transformations [5]. This corresponds to orthogonal symplectic transformations on the columns of $K - \lambda L$. First, we zero $k_{10,4}$ using $k_{10,3}$. The symplectic transformation required to do this introduces non-zero entries in $l_{9,8}$ and $l_{3,4}$ (denoted by $\otimes$ in (5)) which can be zeroed using $l_{8,8}$ and $l_{4,4}$ respectively by means of modified Householder transformations on rows 8 and 9, and 3 and 4 of $K - \lambda L$. The elements $k_{10,3}$ and $k_{10,2}$ can be eliminated in the same way while maintaining the upper triangular structure of L_{22} and the lower triangular structure of L_{11}. If we denote the Householder transformations applied in this step by Q_2 and Z_2, then we have

$$K - \lambda L := Q_2(K - \lambda L)Z_2$$

where the resulting pencil $K - \lambda L$ has the structure

$$\left[\begin{array}{ccccccccccc}
\times & \times & \times & \times & \times & & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & & \times & \times & \times & \times & \times \\[2ex]
\times & \times & \times & \times & \times & & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & & \times & \times & \times & \times & \times \\
\times & 0 & 0 & 0 & \times & & \times & \times & \times & \times & \times
\end{array}\right] - \lambda \left[\begin{array}{ccccccccccc}
\times & 0 & 0 & 0 & 0 & & \times & \times & \times & \times & \times \\
\times & \times & \otimes & 0 & 0 & & \times & \times & \times & \times & \times \\
\times & \times & \times & \otimes & 0 & & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & 0 & & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & & \times & \times & \times & \times & \times \\[2ex]
0 & 0 & 0 & 0 & 0 & & \times & \times & \times & \times & \times \\
0 & 0 & 0 & 0 & 0 & & 0 & \times & \times & \times & \times \\
0 & 0 & 0 & 0 & 0 & & 0 & \otimes & \times & \times & \times \\
0 & 0 & 0 & 0 & 0 & & 0 & 0 & \otimes & \times & \times \\
0 & 0 & 0 & 0 & 0 & & 0 & 0 & 0 & 0 & \times
\end{array}\right] \tag{5}$$

We now apply a Givens symplectic transformation $G_1 = G(1, c, s)$ to appropriate *columns* of $K - \lambda L$ to zero $k_{10,1}$ using $k_{10,6}$. This operation introduces a nonzero entry (denoted by $\otimes$) in $l_{6,1}$ which is zeroed using l_{11} by means of a symplectic Givens transformation $G_2 = G(1, c, s)$ applied to *rows* 1 and 6 of $K - \lambda L$. The resulting pencil

$$K - \lambda L := G_2(K - \lambda L)G_1$$

has the following structure:

$$\begin{bmatrix} \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\[6pt] \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ 0 & 0 & 0 & 0 & \times & \times & \times & \times & \times & \times \end{bmatrix} - \lambda \begin{bmatrix} \times & 0 & 0 & 0 & 0 & \times & \times & \times & \times & \times \\ \times & \times & 0 & 0 & 0 & \times & \times & \times & \times & \times \\ \times & \times & \times & 0 & 0 & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & 0 & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\[6pt] \otimes & 0 & 0 & 0 & 0 & \times & \times & \times & \times & \times \\ 0 & 0 & 0 & 0 & 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \times \end{bmatrix}$$

We have now reduced the appropriate entries in the last row of K_{21} to zero. In order to complete the first step, we need to zero $k_{10,i}, \quad i = 6, 7, 8$. This will allow us to perform the next step without destroying the structure we have already obtained. Also, at the termination of the algorithm, it will give us K_{22} in upper Hessenberg form. Elimination of $k_{10,i}, \quad i = 6, 7, 8$ can be carried out using 2×2 modified Householder transformations. This corresponds to Householder symplectic transformations on the columns of $K - \lambda L$. First, the element $k_{10,6}$ is zeroed using $k_{10,7}$. This results in non-zero entries in $l_{1,2}$ and $l_{7,6}$ (denoted by $\otimes$ in (6)) which can be zeroed using $l_{2,2}$ and $l_{6,6}$ respectively with Householder transformations on rows 1 and 2, and 6 and 7 of $K - \lambda L$. The elements $k_{10,7}$ and $k_{10,8}$ can be zeroed in the same way while maintaining the upper triangular form of L_{22} and the lower triangular form of L_{11}. Denoting the transformations in this operation by Q_3 and Z_3, we have

$$K - \lambda L := Q_3(K - \lambda L)Z_3$$

where Q_3 is orthogonal and Z_3 is orthogonal symplectic. The resulting structure of $K - \lambda L$ is

$$\begin{bmatrix} \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\[6pt] \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\ 0 & 0 & 0 & 0 & \times & 0 & 0 & 0 & \times & \times \end{bmatrix} - \lambda \begin{bmatrix} \times & \otimes & 0 & 0 & 0 & \times & \times & \times & \times & \times \\ \times & \times & 0 & 0 & 0 & \times & \times & \times & \times & \times \\ \times & \times & \times & 0 & 0 & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & 0 & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\[6pt] \otimes & 0 & 0 & 0 & 0 & \times & \times & \times & \times & \times \\ 0 & 0 & 0 & 0 & 0 & \otimes & \times & \times & \times & \times \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \times \end{bmatrix} \tag{6}$$

This completes the first step in the reduction of $K - \lambda L$ to the desired condensed form. We can repeat the procedure to zero appropriate entries in rows 9, 8, and 7 of K, while maintaining the triangular structures of L_{11} and L_{22}. We finally get the following structure for $K - \lambda L$:

$$\left[\begin{array}{ccccc@{\quad}ccccc}
\times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\[6pt]
\times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\
0 & \times & \times & \times & \times & \times & \times & \times & \times & \times \\
0 & 0 & \times & \times & \times & 0 & \times & \times & \times & \times \\
0 & 0 & 0 & \times & \times & 0 & 0 & \times & \times & \times \\
0 & 0 & 0 & 0 & \times & 0 & 0 & 0 & \times & \times
\end{array}\right]
-\lambda
\left[\begin{array}{ccccc@{\quad}ccccc}
\times & 0 & 0 & 0 & 0 & \times & \times & \times & \times & \times \\
\times & \times & 0 & 0 & 0 & \times & \times & \times & \times & \times \\
\times & \times & \times & 0 & 0 & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & 0 & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\[6pt]
\otimes & 0 & 0 & 0 & 0 & \times & \times & \times & \times & \times \\
0 & 0 & 0 & 0 & 0 & 0 & \times & \times & \times & \times \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \times & \times & \times \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \times & \times \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \times
\end{array}\right]$$

If required, further row operations can be performed on rows 1 to 5 of $K - \lambda L$ using Householder transformations to reduce L_{11} to an upper triangular matrix. Also, it is worth mentioning that a similar procedure can be developed to reduce $K - \lambda L$ to a slightly different condensed form:

$$\left[\begin{array}{ccccc@{\quad}ccccc}
\times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\
0 & \times & \times & \times & \times & \times & \times & \times & \times & \times \\
0 & 0 & \times & \times & \times & \times & \times & \times & \times & \times \\
0 & 0 & 0 & \times & \times & \times & \times & \times & \times & \times \\[6pt]
\times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\
0 & \times & \times & \times & \times & \times & \times & \times & \times & \times \\
0 & 0 & \times & \times & \times & \times & \times & \times & \times & \times \\
0 & 0 & 0 & \times & \times & \times & \times & \times & \times & \times \\
0 & 0 & 0 & 0 & \times & \times & \times & \times & \times & \times
\end{array}\right]
-\lambda
\left[\begin{array}{ccccc@{\quad}ccccc}
\times & \times & \times & \times & \times & \times & \times & \times & \times & \times \\
0 & \times & \times & \times & \times & \times & \times & \times & \times & \times \\
0 & 0 & \times & \times & \times & \times & \times & \times & \times & \times \\
0 & 0 & 0 & \times & \times & \times & \times & \times & \times & \times \\
0 & 0 & 0 & 0 & \times & \times & \times & \times & \times & \times \\[6pt]
0 & 0 & 0 & 0 & 0 & \times & \times & \times & \times & \times \\
0 & 0 & 0 & 0 & 0 & 0 & \times & \times & \times & \times \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \times & \times & \times \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \times & \times \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \times
\end{array}\right]$$

ALGORITHM 2.1: *Condensed Form for a Symplectic Pencil*

Given a $2n \times 2n$ symplectic pencil $K - \lambda L$, this algorithm overwrites K and L by matrices QKZ and QLZ respectively with the structure shown in (4). The matrix Q is orthogonal, and the matrix Z is orthogonal symplectic.

```
1. Compute Householder matrices $P_1$, $P_2$, ..., $P_{2n-1}$ such that if $U = P_{2n-1}$, ..., $P_1$,
   and $UL = \begin{bmatrix} L_{11} & L_{12} \\ L_{21} & L_{22} \end{bmatrix}$, then $L_{21} = 0$, and $L_{11}, L_{22} \in \mathbb{R}^{n \times n}$ are lower and upper
   triangular respectively.

2. $K := UK$
   $L := UL$
   $Q := U$
   $Z := I_{2n}$

3. For $i = n : -1 : 2$

      For $j = i - 1 : -1 : 2$
```

```
    Determine a Householder matrix $\bar{V} \in \mathbb{R}^{2\times 2}$ such that
        $\begin{bmatrix} k_{n+i,j-1} & k_{n+i,j} \end{bmatrix} \bar{V} = [0 \;\; *]$
    $K := K\bar{Z}, \quad L := L\bar{Z}$
    $Z := Z\bar{Z}$
    where $\bar{Z} = diag[I_{j-2}, \bar{V}, I_{n-j}, I_{j-2}, \bar{V}, I_{n-j}]$.
    Determine a Householder matrix $\bar{U} \in \mathbb{R}^{2\times 2}$ such that
        $\bar{U} \begin{bmatrix} l_{n+j-1,n+j-1} \\ l_{n+j,n+j-1} \end{bmatrix} = \begin{bmatrix} * \\ 0 \end{bmatrix}$
    $K := \bar{Q}K, \quad L := \bar{Q}L$
    $Q := \bar{Q}Q$
    where $\bar{Q} = diag[I_{n+j-2}, \bar{U}, I_{n-j}]$.
    Determine a Householder matrix $\bar{U} \in \mathbb{R}^{2\times 2}$ such that
        $\bar{U} \begin{bmatrix} l_{j-1,j} \\ l_{j,j} \end{bmatrix} = \begin{bmatrix} 0 \\ * \end{bmatrix}$
    $K := \bar{Q}K, \quad L := \bar{Q}L$
    $Q := \bar{Q}Q$
    where $\bar{Q} = diag[I_{j-2}, \bar{U}, I_{2n-j}]$.
End
Determine constants $c_1$ and $s_1$ with $c_1^2 + s_1^2 = 1$ such that
    $[k_{n+i,1} \;\; k_{n+i,n+1}] \begin{bmatrix} c_1 & s_1 \\ -s_1 & c_1 \end{bmatrix} = [0 \;\; *]$
$K := KG_1, \quad L := LG_1$
where $G_1 = G(1, c, s)$ is a symplectic Givens transformation matrix.
Determine constants $c_2$ and $s_2$ with $c_2^2 + s_2^2 = 1$ such that
    $\begin{bmatrix} c_2 & s_2 \\ -s_2 & c_2 \end{bmatrix} \begin{bmatrix} l_{1,1} \\ l_{n+1,1} \end{bmatrix} = \begin{bmatrix} * \\ 0 \end{bmatrix}$
$K := G_2K, \quad L := G_2L$
where $G_2 = G(1, c, s)$ is a symplectic Givens transformation matrix.
For $j = 1 : i - 2$
    Determine a Householder matrix $\bar{V} \in \mathbb{R}^{2\times 2}$ such that
        $\begin{bmatrix} k_{n+i,n+j} & k_{n+i,n+j+1} \end{bmatrix} \bar{V} = [0 \;\; *]$
    $K := K\bar{Z}, \quad L := L\bar{Z}$
    $Z := Z\bar{Z}$
    where $\bar{Z} = diag[I_{j-1}, \bar{V}, I_{n-j-1}, I_{j-1}, \bar{V}, I_{n-j-1}]$.
    Determine a Householder matrix $\bar{U} \in \mathbb{R}^{2\times 2}$ such that
        $\bar{U} \begin{bmatrix} l_{n+j,n+j} \\ l_{n+j+1,n+j} \end{bmatrix} = \begin{bmatrix} * \\ 0 \end{bmatrix}$
```

```
            $K := \bar{Q}K, \quad L := \bar{Q}L$
            $Q := \bar{Q}Q$
            where $\bar{Q} = diag[I_{n+j-1}, \bar{U}, I_{n-j-1}]$.
            Determine a Householder matrix $\bar{U} \in \mathbb{R}^{2\times 2}$ such that
                $\bar{U}\begin{bmatrix} l_{j,j+1} \\ l_{j+1,j+1} \end{bmatrix} = \begin{bmatrix} 0 \\ * \end{bmatrix}$
            $K := \bar{Q}K, \quad L := \bar{Q}L$
            $Q := \bar{Q}Q$
            where $\bar{Q} = diag[I_{j-1}, \bar{U}, I_{2n-j-1}]$.
        End
    End
```

This algorithm requires about $35n^3$ floating point operations (flops). If it is required to compute a basis for the stable deflating subspace of $K - \lambda L$, then the transformations applied to the columns of $K - \lambda L$ need to be accumulated in Z. This requires approximately $6n^3$ flops.

3. Computation of the Stable Deflating Subspace of $K - \lambda L$

In this section, we propose a "block" shift strategy to reduce the condensed form of the previous section to an upper block triangular form, from which a basis for the stable deflating subspace of $K - \lambda L$ can be obtained directly. In the block triangular form, the $(1,1)$ and $(2,2)$ blocks are $n \times n$ matrices with only stable and unstable eigenvalues respectively. The use of (implicit) single or double shift strategies (as in the QZ algorithm) is not computationally viable because the application of these shifts destroys the structure of the condensed form, and reduction to the form is relatively expensive in an iterative process. However, an approach which applies n shifts simultaneously may be computationally attractive. The problem, of course, is to determine what shifts to apply so as to ensure rapid convergence of the condensed form to the desired upper block triangular form.

One way to determine the n shifts is to use the method proposed in [4] to compute the eigenvalues of $K - \lambda L$. In [4], instead of computing the eigenvalues of $K - \lambda L$ we compute the eigenvalues of a pencil $\tilde{K} - \lambda\tilde{L}$, where $\tilde{K} \equiv KJL^T + LJK^T$, $\tilde{L} \equiv LJL^T$. It can be shown that if λ is an eigenvalue of $K - \lambda L$, then $\lambda + \frac{1}{\lambda}$ is an eigenvalue of $\tilde{K} - \lambda\tilde{L}$. The eigenvalues of $\tilde{K} - \lambda\tilde{L}$ are in turn the same as those of a pencil

$$\bar{K} - \lambda\bar{L} = \begin{bmatrix} Y & W \\ X & Y^T \end{bmatrix} - \lambda \begin{bmatrix} A & 0 \\ 0 & A^T \end{bmatrix}$$

where $X = A^T H - HA$ and $W = FA^T - AF$ are skew-symmetric matrices, and $Y = A^2 + FH + I$. In [4], an algorithm is given for computing the eigenvalues of $K - \lambda L$ by computing the eigenvalues of $\bar{K} - \lambda\bar{L}$. The algorithm uses orthogonal transformations only which preserve the structure of $\bar{K} - \lambda\bar{L}$, and requires approximately $\frac{89}{3}n^3$ flops.

In order to apply block shifts to reduce $K - \lambda L$ to the desired upper block triangular form, we will assume that the eigenvalues of $K - \lambda L$ have already been computed using the method in [4]. A QZ-type algorithm with implicit n (block) shifts for the symplectic pencil $K - \lambda L$ can then be stated as follows:

ALGORITHM 3.1: *QZ-Type Algorithm with Implicit Block Shifts*

Input: A $2n \times 2n$ symplectic pencil $K - \lambda L$, and a set of n unstable eigenvalues λ_i, $i = 1, \ldots, n$, of $K - \lambda L$ computed using the algorithm in [4].

Output: $K - \lambda L := Q(K - \lambda L)Z$ with n implicit shifts corresponding to the λ_i, $i = 1, \ldots, n$, implemented simultaneously. Q is orthogonal and Z is orthogonal symplectic.

1. Find an orthogonal matrix Q and a symplectic orthogonal matrix Z such that $K - \lambda L := Q(K - \lambda L)Z$ is in the condensed form resulting from Algorithm 2.1.

2. $M := L^{-1}K$

   $v^T := e_{2n}^T(M - \lambda_1 I_{2n})(M - \lambda_2 I_{2n}) \cdots (M - \lambda_n I_{2n})$

   where e_{2n}^T is the last row of I_{2n}.

   Determine a symplectic orthogonal matrix Z_1 such that

   $$v^T Z_1 = \alpha e_{2n}^T, \quad \alpha = \| v \| .$$

   $K := KZ_1, \quad L := LZ_1, \quad Z := ZZ_1$

   Determine an orthogonal matrix Q_2 and a symplectic orthogonal matrix Z_2 such that

   $$e_{2n}^T Z_2 = e_{2n}^T;$$

   and

   $$K - \lambda L := Q_2(K - \lambda L)Z_2$$

   is in the condensed form of Algorithm 2.1.

   $K := Q_2 K Z_2, \quad L := Q_2 L Z_2, \quad Q := Q_2 Q, \quad Z := ZZ_2$

In exact arithmetic and with accurate values for λ_i, at the termination of the Algorithm 3.1, we should have the desired upper block triangular structure for $K - \lambda L$. If this is not the case, there is the possibility of deflating the problem. The deflation can be accomplished depending on the entries of K_{21} and K_{22} that have been reduced to zero. These and other numerical issues concerning this algorithm will be discussed in an extended version of this paper. It suffices to mention here that a similar approach has been proposed for Hamiltonian matrices in [6].

It should be noted that the transformation $Q(K - \lambda L)Z$ corresponds to a similarity transformation on $M = L^{-1}K$ by an orthogonal symplectic matrix Z. Also, it can be easily shown that M is a *symplectic matrix* so that this similarity transformation preserves its structure. The condensed form for $K - \lambda L$ in (4) corresponds to a condensed form for M. A similar form for a symplectic matrix is discussed in [9] where a result corresponding to the *Implicit Q Theorem* [5] and an associated QR-type algorithm are stated for the reduction of a symplectic matrix to such a form.

4. Concluding Remarks

In this paper, we have presented an algorithm for reducing a symplectic pencil to a "Hessenberg-type" condensed form using structure preserving orthogonal transformations. Based on this form, we have proposed a QZ-type algorithm for obtaining a block triangular form of the symplectic pencil. This algorithm performs block implementation of multiple shifts. We are currently investigating various numerical and implementation issues concerning this approach. **5. References**

1. T. Pappas, A.J. Laub and N.R. Sandell, "On the numerical solution of the discrete-time algebraic Riccati equation", *IEEE Trans. Autom. Control*, vol. AC-25, pp. 631-641, 1980.
2. P. Van Dooren, "A generalized eigenvalue approach for solving Riccati equations", *SIAM J. Sci. Statist. Comput.*, vol. 2, pp. 121-135, 1981.
3. V. Mehrmann, "A symplectic orthogonal method for single input or single output discrete time optimal quadratic control problems", *SIAM J. Matrix Anal. Appl.*, vol. 9, pp. 221-247, 1988.
4. R.V. Patel, "On computing the eigenvalues of a symplectic pencil", *Lin. Alg. Appl.*, vol. 188,189, pp. 591–611, 1993.
5. G.H. Golub and C. Van Loan, *Matrix Computations*, Johns Hopkins Univ. Press, Baltimore, 1983.
6. G. Ammar and V. Mehrmann, "On Hamiltonian and symplectic Hessenberg forms", *Lin. Alg. Appl.*, vol. 149, pp. 55-72, 1991.
7. C. Van Loan, "A symplectic method for approximating all the eigenvalues of a Hamiltonian matrix", *Lin. Alg. Appl.*, vol. 61, pp. 233-251, 1984.
8. W.-W. Lin, "A new method for computing the closed-loop eigenvalues of a discrete-time algebraic Riccati equation", *Lin. Alg. Appl.*, vol. 96, pp. 157-180, 1987.
9. A. Bunse-Gerstner, R. Byers, and V. Mehrmann, "A chart of numerical methods for structured eigenvalue problems", *SIAM J. Matrix Anal. Appl.*, vol. 13, pp. 419-453, 1992.

Experimental Evaluation of the Gradient Annealing Algorithm

Mohamed Mohandes and Craig Codrington
School of Electrical Engineering, Purdue University
West Lafayette, IN 47907-1285

Abstract

Simulated annealing (SA) is a popular approach to global optimization of functions with multiple local minima. There are a variety of SA algorithms for different types of problems; for continuous optimization, the gradient annealing algorithm (GAA) has been suggested. It was shown that under suitable conditions on the rate of decrease of the temperature and stepsize, the sequence generated by the GAA has a distribution which converges to an equilibrium distribution concentrated on the set of global minima.

In this paper we present an experimental evaluation of the GAA, and also combine it with a local search which is applied either periodically or when the algorithm reaches a lower value than the best minimum found so far. Our simulations show a substantial improvement over the standard GAA.

1. Introduction

Simulated Annealing (SA) was introduced by Kirkpatrick, Gelatt and Vecchi [13] as an approach to solve difficult combinatorial optimization problems. The idea was later substantiated by rigorous mathematical proofs and by successful application to a variety of problems in many areas. Proofs are given in [6,10]. Extensive bibliographies of applications can be seen in [3,17].

Simulated Annealing was later generalized to continuous optimization. One way involves simulating a Markov chain using a generalized Metropolis (or related) method. A large amount of theoretical analysis and practical methodology were developed for this case [1,2,4,15,16]. It was reported that the application of simulated annealing to continuous spaces is more involved than its application to discrete problems. In particular, the choice of the random step is more subtle, since the optimal stepsize and direction are not known in advance. A very small stepsize leads to inefficient exploration of the problem space, while a very large stepsize leads to frequent rejections. Also, in high-dimensional problems, where long narrow valleys are common, steps perpendicular to the valley axis are generally not productive. There have been attempts to overcome these problems but in general, the performance of this method remains questionable.

Another type of annealing algorithm for continuous optimization has been proposed. It involves modifying gradient-type search algorithms. We refer to this method as the Gradient Annealing Algorithm (GAA). Let $U(\cdot)$ be a smooth cost function on R^d. A standard gradient algorithm for finding a local minimum of $U(\cdot)$ (a global minimum if $U(\cdot)$ is convex) is given by:

$$z_{k+1} = z_k - \mu \nabla U(z_k)$$

where μ is a stepsize parameter. The gradient annealing algorithm for finding a global minimum of $U(\cdot)$ is given by:

$$x_{k+1} = x_k - \mu \nabla U(x_k) + \sqrt{2T\mu}\, w_k$$

where $\{w_k\}$ is a white Gaussian noise sequence and T is a temperature parameter which is slowly decreased as the algorithm proceeds.

Some theoretical analysis has been developed for the gradient annealing algorithm [7,8,14]. However, there has been no experimental evaluation that we are aware of. On the other hand, there may be some hope of using the GAA for high-dimensional problems with smooth well-behaved cost functions, as it attempts to exploit the smoothness by its use of derivatives. The goal of this paper is to present an experimental evaluation of the gradient annealing algorithm. Simulated annealing algorithms in general are inefficient because they frequently enter the region of attraction of a local minimum and leave it without actually reaching that minimum. In this paper we try to overcome this inefficiency by combining the gradient annealing algorithm with a local search done either periodically or when the algorithm reaches a lower function value than the best found so far. Our simulations show a substantial improvement in performance over the standard GAA.

2. The Gradient Annealing Algorithm

Let $U(\cdot)$ be a smooth cost function (at least C^2) on R^d. We wish to find an element of R^d which minimizes $U(\cdot)$. The GAA (with a fixed stepsize μ) is given by the following stochastic recursion

$$x_{k+1} = x_k - \mu \nabla U(x_k) + \sqrt{2T\mu}\, w_k$$

where $\{w_k\}$ is a standard d-dimensional white Gaussian noise sequence, artificially added in (via Monte Carlo simulation) to try to avoid getting trapped in local minima, and T is the temperature. The temperature T (and possibly the stepsize μ) are slowly decreased as k gets large to simulate the annealing process. For fixed temperature T and small stepsize μ the process $\{x_k\}$ nearly has a Gibbs equilibrium distribution with density function $\pi(T,.)$, and as the temperature T tends to zero we get $\pi(T,.)$ converging to $\pi^*(.)$ which is concentrated on the global minima of $U(\cdot)$. Under some conditions on the rate of temperature and stepsize decrease, $\{x_k\}$ remains near the equilibrium distribution and also concentrates on the global minimum for large k [8,9,14].

2.1 The Algorithm

In this section we describe the algorithm that we have used in our experiments. Let $U(x)$ be a smooth cost function (C^2) on a bounded subset $S \subseteq R^d$. Let U^* be the global minimum of $U(\cdot)$ on S. Input parameters:
T_0: initial temperature; T_F: final temperature; N: temperature length (# iterations at each temperature); ρ: geometric ratio in temperature schedule; μ stepsize;
ε: precision parameter (termination threshold).

```
1.  Set $T = T_0$, $j = 0$.
2.  Select $x_0$ randomly from S and compute $U_0 = U(x_0)$.
3.  If $|U_k - U^*| < \varepsilon$ or $(T < T_F)$ stop; otherwise
4.  Set $T_j = \rho^j T_0$.
5.  Do the following N times
        $x_{k+1} = x_k - \mu\nabla U(x_k) + \sqrt{2T_j\mu}\, w_k$
        set $x_{k+1} = x_k$
6.  $j = j + 1$
7.  Go to step 3.
```

In step 5, if any component x^i_{k+1} steps over the boundary of S, we use backtracking to project it back inside S.

2.2 Optimizing the Parameter Settings

As in all other annealing algorithms, the GAA requires that a variety of parameters be specified; the initial temperature T_0, the final temperature T_F (the termination condition for the algorithm), the temperature length N (# iterations at each temperature), the stepsize, and the temperature ratio (the ratio of temperature decrease). For the generalized Metropolis method, the acceptance probability (the probability of accepting uphill moves) can help determine the initial and the final temperatures. However, in the GAA all steps are accepted. Consequently, there is no systematic method to determine the temperature range. In future research we will find such a method by finding a match between the generalized Metropolis and the GAA. Here, we have to depend on extensive experiments to find suitable values for each of the test functions. Due to the large number of parameters, we cannot investigate all possible combinations. We follow [12] by studying only one factor at a time in hopes of isolating its effects. In our simulations, we terminate the algorithm when we are within a specified distance ε from the the global minimum or when the total number of iterations exceeds a specified number M, rather than when T_F is reached. Thus we need only consider T_0, μ, N, and ρ. We fix three of these parameters and change the fourth over a reasonable range and then repeat the process for each of the other parameters. The effect of different parameter settings are shown in figures 1-4. The other test functions are reported in tabular form (tables 1-6) because for some of the parameters, not all runs converged to the global minimum and we have to associate the percentage of convergent runs with the performance of the algorithm.

3. The Combined Algorithm

It was shown that under suitable conditions on the rate of decrease of the temperature and stepsize, the sequence generated by the GAA has a distribution which converges to an equilibrium distribution concentrated on the set of global minima. However, this rate is far too slow to be used practically and we must stop the algorithm short of the limiting distribution. Thus throughout the annealing process the algorithm may enter and leave the region of attraction of the global minimum without ever actually reaching it. Brooks and Verdini [2] addressed this problem for the generalized Metropolis method by first locating the regions of attraction of all minima and then carrying out a local search from each one. We propose to increase the efficiency of the Gradient Annealing Algorithm by interleaving a local search with the annealing process. The local search is conducted either after a pre-specified number of steps of the annealing process, or when the algorithm reaches a lower function value than the best found so far. The combined algorithm is summarized below:

1. Start at a randomly selected initial point x_0.
2 Apply the local search to find the nearest local minimum.
3. Run the GAA until either
    - the function value drops below the previous minimum or
    - for a specified number of iterations.
4. Goto step 2.

The algorithm stops when the function value is within a specified distance of the global minimum or after a specified number of trials and the best minimum found is reported.

4. Results

Dixon and Szego [5] invited a number of researchers in the field of global optimization to apply their own algorithms to a set of standard test functions. We will consider these test functions individually and compare our results to theirs. As in [5], we measure performance in terms of the number of function evaluations and standard unit times to convergence, where the standard unit time is defined as the time required for 1000 evaluations of the Shekel-5 function. However, as the GAA uses the gradient, there is a gradient evaluation for every function evaluation. As in all stochastic optimization methods, the GAA's performance depends on the initial point and on the path followed. Our performance criterion is the mean number of iterations to convergence when the GAA is run from 500 different initial points.

4.1 Goldstein and Price Function

This is a two dimensional function with four minima, one of which is global. The function is given by

$$U(x,y) = \left[1 + (x + y + 1)^2 \left(19 - 14x + 3x^2 - 14y + 6xy + 3y^2\right)\right] \times$$
$$\left[30 + (2x - 3y)^2 \left(18 - 32x + 12x^2 + 48y - 36xy + 27y^2\right)\right]$$

Region of interest: $S = \{(x,y): -2 \leq x, y \leq 2\}$

Figure 1 shows the effect of the initial temperature on the performance of the algorithm. We note that if the initial temperature is too small, the algorithm converges to a local minimum and will never escape to the global one. Large values of T_0 make many iterations at high temperature unfruitful, and consequently the total number of iterations increases. Figure 2 shows the effect of the temperature length when other parameters are fixed. Small values of N lead to a rapid quenching process and the algorithm's performance is similar to a local optimization algorithm, while very large N cause inefficiency at high temperature because the algorithm jumps to and from the area of the global minimum freely. Figure 3 shows the effect of the stepsize μ; it has similar effect to that of T_0. Figure 4 shows the effect of the temperature ratio. As in crystal formation, a rapid decrease in the temperature results in convergence to a local minimum. Finally, Figure 5 shows the varying number of iterations to convergence for 500 different random initial points. The number of iterations ranged from 5 to 2500 with the majority near a few hundred. A suitable set of parameters is used to report the performance of the GAA on this function in tables 7 and 8.

4.2 Shekel's family (SQRN-5, SQRN-7, SQRN-10)

This family of four dimensional functions has 5, 7, and 10 minima respectively including one global minimum for each. The functions are given by:

$$U(x) = \sum_{i=1}^{m} \frac{1}{(x-a_i)^T(x-a_i)+c_i}$$

where $x = (x_1, x_2, ..., x_n)$, $a_i = (a_{i1}, a_{i2}, ..., a_{in})$, and $c_i > 0$.

Region of interest: $0 \leq x_j \leq 10 \quad j = 1,...,n.$

The values of a and c can be found in [5]. It has been reported that this set of functions cause difficulties for simulated annealing methods in general [2,16]. This is due to the fact that the minima are deep and have small regions of attraction, causing the algorithm either to step over them or be trapped in the first one encountered. The GAA also suffers from these difficulties. Due to the fact that not all runs converged to the global minimum after 20,000 iterations, we report the percentage of runs which converge to the global minimum for each setting of the parameters. We report the results in tables 1, 2, and 3; the performance of the GAA on this set of functions will be presented in tables 7 and 8.

4.3 Hartman's family (H3, H6)

This set includes one three dimensional function with five minima, one of which is global (see [2]), and one six dimensional function with four minima. The functions are given by:

$$U(x) = -\sum_{i=1}^{m} c_i \exp\left(-\sum_{j=1}^{n} a_{ij}(x_j - p_{ij})^2\right)$$

where $x = (x_1, x_2, ..., x_n)$, $p_i = (p_{i1}, p_{i2}, ..., p_{in})$, $a_i = (a_{i1}, a_{i2}, ..., a_{in})$.

p_i is the approximate location of the i'th local minimum, a_i is proportional to the eigenvalues of the hessian matrix at the i'th local minimum, and c_i is the depth of the i'th local minimum.

Region of interest: $0 \leq x_j \leq 1 \quad j = 1,...,m.$

The values of a, p, and c can be found in [5]. The parameter settings are shown in tables 4 and 5 and the performance of the GAA on this set of functions is reported in tables 7 and 8.

4.4 Branin (RCOS)

This two dimensional function has three minima, all global. The function is given by:

$$U(x,y) = a\left(y - bx^2 + cx - d\right)^2 + e\,(1 - f)\cos(x) + e$$

where: $a = 1$, $d = 6$, $e = 10$, $b = 5.1\left(4\pi^2\right)$, $c = \frac{5}{\pi}$, $f = \frac{1}{8\pi}$.

Region of interest: $S = \left\{(x,y):\ -5 \leq x \leq 10,\ 0 \leq y \leq 15\right\}$.

Table 6 shows the results for various settings of the parameters. It reveals that for this function, the performance of the GAA is not unduly sensitive to the parameter settings.The performance of the GAA is shown in tables 7 and 8.

5. Conclusions

We have experimentally evaluated the Gradient Annealing Algorithm, and demonstrated its viability for continuous optimization. It compares favorably with many standard global optimization algorithms, despite the overhead of computing the gradient. Performance improved substantially when the GAA was augmented with a local search strategy, done either periodically or when a lower function value is reached. Not only did it reduce the number of iterations required, it also consistently located the global minimum even when the standard GAA and other annealing algorithms failed to do so. These performance gains are realized even for simple local search strategies such as gradient descent, as we have used; we expect that more sophisticated local search algorithms will yield an even greater advantage to the combined method. Furthermore, we expect that interleaving a local search with most other simulated annealing algorithms will improve their performance as well.

Acknowledgment: The first author thanks Professor Hanis Thompson for his support.

LIST OF REFERENCES

[1] Bohachevsky, I.O., Johnson, M.E.and Stein, M.L. "Generalized simulated annealing for function optimization," Technometrics, Vol. 28, No. 3, pp. 209-217, August 1986.

[2] Brooks, D.G. and Verdini, W.A. "Computational experience with generalized simulated annealing over continuous variables," Amer. J. Math. and Manag. Science, Vol. 8, No. 3&4, pp. 425-449, 1988.

[3] Collins, N.E., Eglese, R.W., and Golden, B.L. "Simulated annealing: an annotated bibliography". Report no. 88-019, College of Business and Management, University of Maryland, College Park. 1988.

[4] Corona, A., Marchesi, M., Martini, C, and Ridella, S. "Minimizing multimodal functions of continuous variables with the "simulated annealing" algorithm," ACM Trans. Math. Software, Vol. 13, No. 3, pp. 262-280, 1987.

[5] Dixon, I.C.W. and Szego, G.P. "Towards global optimization 2," North-Holland Publishing Company, 1978.

[6] Gelfand, S.B. and Mitter, S.K. "Analysis of simulated annealing for optimization in Proc. 24th Conf. on Decision and Control, Ft. Lauderdale, Fla. pp. 779-786, 1985.

[7] Gelfand, S.B. and Mitter, S.K. "Simulated annealing type algorithms for multivariate optimization," Algorithmica, Vol. 6, No. 3, pp. 419-436, May 1991.

[8] Gelfand, S.B. and Mitter, S.K. "Recursive stochastic algorithms for global optimization in R^d," SIAM J. Control and Optimiz., Vol. 29, No. 5, pp. 999-1018, September 1991.

[9] Gelfand, S.B. and Mitter, S.K. "Metropolis-type annealing algorithms for global optimization in R^d" SIAM J. Control and Optimiz., November 1992.

[10] Geman, S. and Geman, D. "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," IEEE Trans. PAMI., Vol. 6, pp. 721-741, 1984.

[11] Hajek, B. "Cooling Schedules for optimal annealing," Mathematics of Oper. Res., Vol. 13, pp. 311-329, 1988.

[12] Johnson, D.S., Aragon, C.R., McGeoch, L.Y. and Schevon, C. "Optimization by simulated annealing: an experimental evaluation: part I, graph partitioning," Oper. Res., Vol. 37, No. 6, pp. 865-892, November-December 1989.

[13] Kirkpatrick, S., Gelatt, C.D. jr. and Vecchi, M.P. "Optimization by simulated annealing," Science Vol. 220, pp. 671-680, 1983.

[14] Kushner, H.J. "Asymptotic global behavior for stochastic approximation and diffusions with slowly decreasing noise effects: global minimization via Monte Carlo," SIAM J. Appl. Math., Vol. 47, pp. 169-185, 1987.

[15] Press, W.H. and Teukolsky, S.A. "Simulated annealing optimization over continuous spaces," Computers in Physics, pp. 426-429, Jul/Aug 1991.

[16] Vanderbilt, D. and Louie, S.G. "A Monte Carlo simulated annealing approach to optimization over continuous variables," J. Comput. Phys, Vol. 56, pp. 259-271, 1984.

[17] Van Laarhoven, P.J.M. and Aarts, E.H.L. "Simulated annealing: theory and practice". Kluwer Academic Publishers, Dordrecht, The Netherlands. 1987.

Figure 1: Ave # of iterations to convergence vs initial T

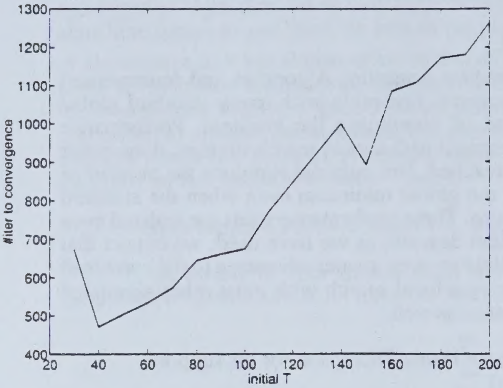


Figure 2: # iterations to convergence vs temperature lengt

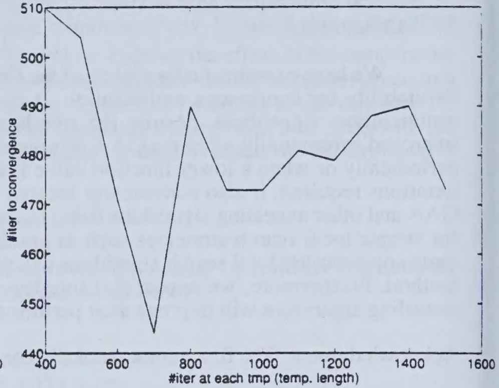


Figure 3: # iterations to convergence vs MU

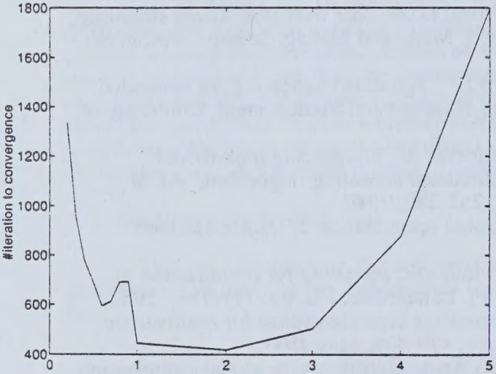


Figure 4: # iterations to convergence vs temperature ratio

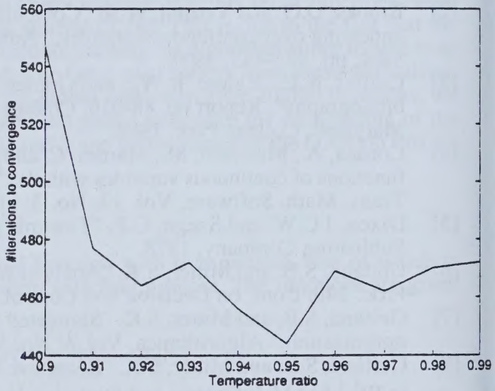
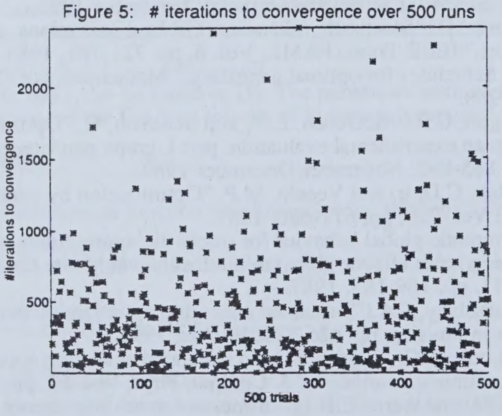


Figure 5: # iterations to convergence over 500 runs



<table>
<tr><td colspan="9">Effect of initial temperature T_0</td></tr>
<tr><td>T_0</td><td>.3</td><td>.4</td><td>.5</td><td>.6</td><td>.7</td><td>.8</td><td>.9</td><td>1</td></tr>
<tr><td>#iter</td><td>6035</td><td>6658</td><td>7700</td><td>8603</td><td>9775</td><td>10892</td><td>12037</td><td>13479</td></tr>
<tr><td>%conv.</td><td>55</td><td>65</td><td>73</td><td>75</td><td>68</td><td>58</td><td>38</td><td>30</td></tr>
<tr><td colspan="9">Effect of stepsize μ</td></tr>
<tr><td>μ</td><td>.01</td><td>.015</td><td>.02</td><td>.025</td><td>.03</td><td>.035</td><td>.04</td><td>.045</td></tr>
<tr><td>#iter</td><td>9099</td><td>8149</td><td>7347</td><td>7755</td><td>8305</td><td>8424</td><td>8800</td><td>10142</td></tr>
<tr><td>%conv.</td><td>48</td><td>64</td><td>77</td><td>92</td><td>91</td><td>80</td><td>80</td><td>73</td></tr>
<tr><td colspan="9">Effect of temperature length N</td></tr>
<tr><td>N</td><td>1400</td><td>1600</td><td>1800</td><td>2000</td><td>2200</td><td>2400</td><td>2600</td><td>2800</td></tr>
<tr><td>#iter</td><td>7497</td><td>7550</td><td>7752</td><td>7695</td><td>7597</td><td>7913</td><td>7542</td><td>7690</td></tr>
<tr><td>%conv.</td><td>72</td><td>74</td><td>81</td><td>75</td><td>78</td><td>77</td><td>80</td><td>77</td></tr>
<tr><td colspan="9">Effect of temperature ration ρ</td></tr>
<tr><td>ρ</td><td>.92</td><td>.93</td><td>.94</td><td>.95</td><td>.96</td><td>.97</td><td>.98</td><td>.99</td></tr>
<tr><td>#iter</td><td>4999</td><td>4887</td><td>5222</td><td>5620</td><td>5433</td><td>5868</td><td>6093</td><td>6148</td></tr>
<tr><td>%conv.</td><td>85</td><td>90</td><td>88</td><td>93</td><td>90</td><td>95</td><td>92</td><td>95</td></tr>
</table>

Table 1 Parameter settings for Shekel-5

<table>
<tr><td colspan="9">Effect of initial temperature T_0</td></tr>
<tr><td>T_0</td><td>.3</td><td>.4</td><td>.5</td><td>.6</td><td>.7</td><td>.8</td><td>.9</td><td>1</td></tr>
<tr><td>#iter</td><td>4207</td><td>5876</td><td>7164</td><td>9330</td><td>11384</td><td>12221</td><td>12951</td><td>13923</td></tr>
<tr><td>%conv.</td><td>72</td><td>89</td><td>93</td><td>81</td><td>69</td><td>49</td><td>28</td><td>20</td></tr>
<tr><td colspan="9">Effect of stepsize μ</td></tr>
<tr><td>μ</td><td>.02</td><td>.022</td><td>.024</td><td>.026</td><td>.028</td><td>.03</td><td>.032</td><td>.034</td></tr>
<tr><td>#iter</td><td>6802</td><td>6876</td><td>7227</td><td>7435</td><td>7487</td><td>7165</td><td>7784</td><td>8301</td></tr>
<tr><td>%conv.</td><td>83</td><td>88</td><td>90</td><td>89</td><td>92</td><td>93</td><td>90</td><td>90</td></tr>
<tr><td colspan="9">Effect of temperature length N</td></tr>
<tr><td>N</td><td>1700</td><td>1800</td><td>1900</td><td>2000</td><td>2100</td><td>2200</td><td>2300</td><td>2400</td></tr>
<tr><td>#iter</td><td>5222</td><td>5398</td><td>5591</td><td>5639</td><td>5389</td><td>5876</td><td>5544</td><td>5678</td></tr>
<tr><td>%conv.</td><td>84</td><td>84</td><td>86</td><td>88</td><td>89</td><td>89</td><td>88</td><td>85</td></tr>
<tr><td colspan="9">Effect of temperature ratio ρ</td></tr>
<tr><td>ρ</td><td>.92</td><td>.93</td><td>.94</td><td>.95</td><td>.96</td><td>.97</td><td>.98</td><td>.99</td></tr>
<tr><td>#iter</td><td>4944</td><td>5118</td><td>5346</td><td>5389</td><td>5702</td><td>6223</td><td>6321</td><td>6337</td></tr>
<tr><td>%conv.</td><td>80</td><td>81</td><td>84</td><td>89</td><td>87</td><td>80</td><td>93</td><td>93</td></tr>
</table>

Table 2 Parameter settings for Shekel -7

Effect of initial temperature T_0								
T_0	.3	.4	.5	.6	.7	.8	.9	1
#iter	3875	5472	7409	9319	11764	12381	13030	13541
%conv.	79	90	91	87	73	48	35	20
Effect of stepsize μ								
μ	.01	.02	.03	.04	.05	.06	.07	.08
#iter	8118	5983	5472	6741	8528	10197	10998	10581
%conv.	67	82	89	92	87	73	60	43
Effect of temperature length N								
N	2000	2100	2200	2300	2400	2500	2600	2700
#iter	5381	5472	5602	5235	5620	5403	5462	5283
%conv.	89	90	89	89	92	90	89	90
Effect of temperature ratio ρ								
ρ	.92	.93	.94	.95	.96	.97	.98	.99
#iter	4999	4887	5222	5620	5433	5868	6093	6148
%conv.	85	90	88	93	90	95	92	95

Table 3 Parameter settings for Shekel -10

Effect of initial temperature T_0								
T_0	.1	.2	.3	.4	.5	.6	.7	.8
#iter	644	865	1234	1564	1820	2218	2286	2521
%conv.	96	100	100	100	100	100	100	100
Effect of stepsize μ								
μ	.010	.015	.020	.025	.030	.035	.040	.045
#iter	311	425	518	680	886	1161	1429	1540
%conv.	99	100	100	100	98	93	85	71
Effect of temperature length N								
N	80	100	120	140	160	180	200	220
#iter	295	298	328	330	319	316	303	309
%conv.	99	98	98	99	99	99	100	100
Effect of temperature ratio ρ								
ρ	.65	.70	.75	.80	.85	.90	.95	.99
#iter	262	251	265	277	287	313	303	339
%conv.	89	90	94	95	96	99	100	100

Table 4 Parameter settings for Hartman-3

Effect of initial temperature T_0

T_0	.01	.02	.03	.04	.05	.06	.07	.08
#iter	2039	8635	6332	5839	5417	5565	6386	6742
%conv.	9	64	76	90	95	93	93	89

Effect of stepsize μ

μ	.07	.08	.09	.10	.11	.12	.15	.20
#iter	6369	5913	5640	5417	5517	6381	6844	8415
%conv.	90	95	95	96	94	94	90	75

Effect of temperature length N

N	1000	1100	1200	1300	1400	1500	1600	1700
#iter	5012	4841	4686	4914	5232	5093	5281	5468
%conv.	82	85	87	91	92	95	92	90

Effect of temperature ratio ρ

ρ	.65	.70	.75	.80	.85	.90	.95	.99
#iter	1776	1989	2165	2872	3646	4482	5093	6148
%conv.	57	51	54	60	74	83	95	95

Table 5 Parameter settings for Hartman -6

Effect of initial temperature T_0

T_0	.3	.4	.5	.6	.7	.8	.9	1
#iter	3398	3455	3531	3408	3437	3307	3436	3346
%conv.	99	99	100	100	99	99	99	99

Effect of stepsize μ

μ	.0001	.0005	.001	.0015	.002	.0025	.003	.0035
#iter	107	3086	3307	4294	5021	4712	4666	3133
%conv.	33	92	100	98	98	97	97	98

Effect of temperature length N

N	40	60	80	100	120	140	160	180
#iter	3310	3397	3427	3307	3298	3344	3381	3540
%conv.	99	99	99	100	100	100	100	99

Effect of temperature ratio ρ

ρ	.50	.60	.70	.80	.90	.95	.97	.99
#iter	3677	3449	3275	3296	3411	3307	3311	3277
%conv.	99	99	99	99	99	100	99	99

Table 6 Parameter settings for Branin

Method	SQRN 5	SQRN 7	SQRN 10	H 3	H 6	G.P.	RCOS
Bremmermann	340L	1700L	500L	505L	L	210L	250
Mod Bremm.	375L	405L	336L	420L	515	300	160
Zilinskas	L	12121L	8892L	8641	-	-	5129
Gomulka/Branin	5500	5020	4860	-	-	-	-
Torn	3679	3606	3974	2584	3447	2499	1558
Gomulka/Torn	6654	6084	6144	-	-	-	-
Gomulka/V.M.	7085	6684	7352	6766	11125	1495	1318
Price	3800	4900	4400	2400	7600	2500	1800
Fagiuoli	2514	2519	2518	513	2916	158	1600
De Biase/Frontini	620	788	1160	732	807	378	597
Mockus	1174	1279	1209	513	1232	362	189
GAA	7677(.77)	5876(.89)	5472(.9)	303	5093(.95)	416	3307
Modified GAA	1695	1698	1784	187	206	188	615

Table 7 Performance Comparison, number of function evaluations. All numbers are averages. L indicates a local minimum was found. The parentheses indicate the fraction of runs in which the GAA found the global minimum.

Method	SQRN 5	SQRN 7	SQRN 10	H 3	H 6	G.P.	RCOS
Bremmermann	1 L	8 L	17 L	2 L	L	.5 L	1 L
Mod Bremm.	1.5 L	1.5 L	2 L	2 L	3	.7	.5
Zilinskas	L	282 L	214L	175	-	-	80
Gomulka/Branin	9	8.5	9.5	-	-	-	-
Torn	10	13	15	8	16	4	4
Gomulka/Torn	17	15	20	-	-	-	-
Gomulka/V.M.	19	23	23	17	48	2	3
Price	14	20	20	8	.46	3	4
Fagiuoli	7	9	13	5	100	.7	5
De Biase/Frontini	23	20	30	16	21	15	14
Mockus	96	258	1059	117	255	14.7	1.5
GAA	36 (.77)	33 (.89)	51 (.9)	1.7	14.3 (.95)	.5	4.3
Modified GAA	7.8	8.2	16	.5	.4	.2	.6

Table 8 Performance Comparison in standard unit times where the standard time unit is thetime for 1000 evaluations of the Shekel-5 function.

Least Squares Approximate Solution of Overdetermined Sylvester Equations

A. Scottedward Hodel[1]

In this paper, we address the least-squares solution of minimizations of the form

$$\min_X \|AXB' + CXD' + F\|_F \tag{1}$$

where A, B, C and $D \in \mathrm{I\!R}^{n \times k}$, $X \in \mathrm{I\!R}^{k \times k}$, $F \in \mathrm{I\!R}^{n \times n}$, and $k \ll n$. Minimizations of this form arise in algorithms for low-rank approximate solution of Lyapunov and Sylvester equations [2]. The dimension of the LS minimization (1) may be reduced by applying $n \times 2k$ QR factorizations $Q_1 \begin{bmatrix} C_1 & A_1 \\ 0 & A_2 \end{bmatrix} = \begin{bmatrix} C & A \end{bmatrix}$ and $Q_2 \begin{bmatrix} B_1 & D_1 \\ 0 & D_2 \end{bmatrix} = \begin{bmatrix} B & D \end{bmatrix}$. Then $X \in \mathrm{I\!R}^{k \times k}$ minimizes (1) if and only if X minimizes $\left\| \begin{bmatrix} A_1XB_1' + C_1XD_1' \\ A_2XB_1' \\ C_1XD_2' \end{bmatrix} + \begin{bmatrix} \hat{F}_{11} \\ \hat{F}_{21} \\ \hat{F}_{12} \end{bmatrix} \right\|_F$ where $\hat{F}_{11}$, $\hat{F}_{12}$, and $\hat{F}_{21}$ are extracted from the appropriate blocks of $Q_1'FQ_2$. Numerical solution of the minimization (1) may proceed by application of the conjugate gradient (CG) algorithm or the generalized conjugate gradient (GCG) algorithm [1] to the reduced dimension minimization. The corresponding GCG algorithm is as follows.

```
1. $X_{-1} = X_0 = 0$, $j = 0$, $R_{11}^{(0)} = \hat{F}_{11}$, $R_{21}^{(0)} = \hat{F}_{21}$, $R_{12}^{(0)} = \hat{F}_{12}$,
       $R_0 = A_1'\hat{F}_{11}B_1 + C_1'\hat{F}_{11}D_1 + A_2'\hat{F}_{21}B_1 + C_1'\hat{F}_{12}D_2$
2. while $R_j \neq 0$
    (a) $j = j+1$; Solve via $QZ$: $0 = A_1'T_jB_1 + C_1'T_jD_1 - \left(A_2'R_{21}^{(j-1)}B_1 + C_1'R_{12}^{(j-1)}D_2\right)$
        and $0 = A_1Z_jB_1' + C_1Z_jD_1' + (R_{11}^{(j-1)} + T_j)$ for $T_j$ and $Z_j$.
    (b) Compute $W_{11}^{(j)} = A_1Z_jB_1' + C_1Z_jD_1'$, $W_{21}^{(j)} = A_2Z_jB_1'$, and $W_{12}^{(j)} = C_1Z_jD_2'$,
        and let $W_j = \begin{bmatrix} \mathrm{vec}(W_{11}^{(j)})' & \mathrm{vec}(W_{21}^{(j)})' & \mathrm{vec}(W_{12}^{(j)})' \end{bmatrix}'$.
    (c) $\gamma_j = \left(-\mathrm{vec}(Z_j)'\mathrm{vec}(R_{j-1})\right) / \left(\mathrm{vec}(W_j)'\mathrm{vec}(W_j)\right)$
    (d) $\omega_j = 1$ if $j = 1$, else $\omega_j = \left(1 - \left(\gamma_j(\mathrm{vec}(Z_j)'r_{j-1})\right) / \left(\omega_{j-1}\gamma_{j-1}(\mathrm{vec}(Z_{j-1})'r_{j-2})\right)\right)^{-1}$
    (e) $X_j = X_{j-2} + \omega_j(\gamma_jZ_j + X_{j-1} - X_{j-2})$.
    (f) Compute residuals $R_{11}^{(j)} = A_1X_{j-1}B_1' + C_1X_{j-1}D_1' + \hat{F}_{11}$, $R_{21}^{(j)} = A_2X_{j-1}B_1' + \hat{F}_{21}$, $R_{12}^{(j)} = C_1X_{j-1}D_2' + \hat{F}_{12}$,and $R_j = A_1'R_{11}^{(j)}B_1 + C_1'R_{11}^{(j)}D_1 + A_2'R_{21}^{(j)}B_1 + C_1'R_{12}^{(j)}D_2$.
3. end while
4. $X = X_j$
```

Convergence behavior of this algorithm is treated in [1]. The GCG approach solves minimizations (1) that arise in [2] in $2k$ iterations, or $O(k^4)$ flops.

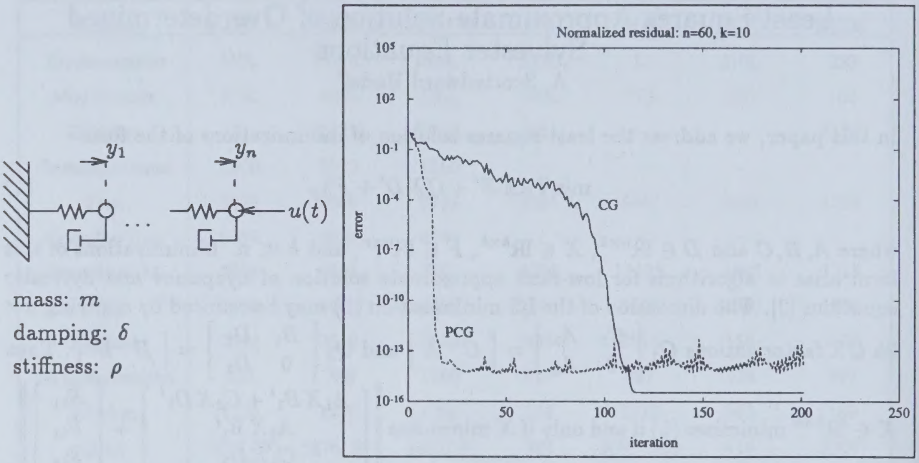


Figure 1: Mass-spring-dashpot system and algorithm results

Example Consider the lumped mass-spring-damper model shown in Figure 1. The first order dynamic model of the system is $\dot{x} = Ax + Bu$ where $A = \begin{bmatrix} 0 & I_n \\ (\rho/m)A_{21} & -(\delta/m)I_n \end{bmatrix}$, A_{21} is a tridiagonal matrix with diagonal elements $(-2, \ldots, -2, -1)$ and ones on the off diagonal entries, and $B = \begin{bmatrix} 0 & \cdots & 0 & 1 \end{bmatrix}'$. Let $k = 10$ and let V be a matrix satisfying $\operatorname{span}(V) = \operatorname{span}\left(\begin{bmatrix} B & AB & \cdots & A^{k-1}B \end{bmatrix}\right)$, $V'V = I$. A solution X was sought to the minimization $\min_{X \in \mathbb{R}^{k \times k}} \|(AV)XV' + VX(A'V') + BB'\|_F$ Computed residuals $\|R_j\|$ are shown in Figure 1 for parameters $\delta = 0.1$, $\rho = 1$, and $m = 1$.

Acknowledgement The author wishes to thank Gene Golub for suggeting the overdetermined Sylvester equation problem.

References

[1] Paul Concus, Gene H. Golub, and Dianne P. O'Leary. A generalized conjugate gradient method for the numerical solution of elliptic partial differential equations. In J. R. Bunch and D. J. Rose, editors, *Sparse Matrix Computations*, pages 309–332. Academic Press, 1976.

[2] D. Y. Hu and L. Reichel. Krylov subspace methods for the Sylvester equation. *Linear Algebra and Its Applications*, 172:283–313, 1992.

Uplink Performance of a CDMA Wireless Communications System

Li Fung Chang

Radio Research Department
Bellcore
331 Newman Springs Rd.,
Red Bank, NJ 07701, U. S. A.

Abstract

In recent years, Code-Division-Multiple Access (CDMA) technology has been widely studied and considered for mobile/personal digital communications. This paper reviews several performance aspects of the uplink of a CDMA wireless communications system. Several factors that influence the uplink performance are addressed. Factors such as, effectiveness of feedback power control, path diversity, Walsh word-by-word interleaving vs. bit-by-bit interleaving in case of orthogonal signaling with noncoherent combining, and coherent detection with reference-symbol based channel estimation vs. noncoherent detection using Walsh orthogonal signals are discussed in this paper.

1 INTRODUCTION

Code-division multiple access using spread spectrum direct sequence (for convenience, herein called CDMA) has been considered as an alternative digital wireless technology for mobile and personal radio communications.[1,2,3] For CDMA, a unique binary spreading sequence is assigned to each user, and all users share the same frequency spectrum. Since all users share the same frequency spectrum, the total amount of interference power received at a receiver is determined by all the other simultaneous users if the sequences are not orthogonal. The near-far problem is a result of some transmitters near a receiver generating overwhelming interference for other transmitters far from the receiver which experience large transmission attenuation. Feedback power control is used so that a base station will receive the same power from each portable transmitter, thereby solving the "near-far" problem at the uplink receiver.

This paper reviews several performance aspects on the uplink of a CDMA wireless communications system with feed-back power control that have been previously presented by the author and other researchers. We focus on the link level performance where orthogonal signaling with noncoherent detection is employed. Performance of the channel using error correcting coding with bit-by-bit or word-by-word interleaving is shown. The effect of the use of multiple correlators per antenna in various multipath delay-spread environments is discussed.

2. Transmitter/Receiver Model

In this section, the structure of the transmitter/receiver for CDMA uplinks with feedback power control is discussed. Orthogonal signaling using 64-ary Walsh codes and noncoherent detection are considered.(this does not imply orthogonality among users) Walsh word-by-word interleaving and conventional bit-by-bit interleaving techniques are described.

2.1. Transmitter

The transmitter includes a channel coder, Walsh mapper, interleaver, and PN sequence generator. A rate 1/3, constraint length 9 convolutional code with three generator polynomials ($g_1(x)$, $g_2(x)$, $g_3(x)$) is used, where $g_1(x) = x^8+x^6+x^5+x^3+x^2+x+1$ (i.e. in octal form g_1=557), g_2=663 and g_3=711. The information bits are first encoded by using a rate 1/3 convolutional channel coder. For the bit-by-bit interleaving technique: the coded bits are interleaved by using a block interleaver; the interleaved coded bits are then organized into six bit groups and then mapped into Walsh codewords (Figure 1(a)). For the Walsh word-by-word interleaving technique: the coded bits are first organized into six bit groups; the groups are mapped into Walsh codewords and then interleaved word-by-word [4,5](Figure 1(b)).

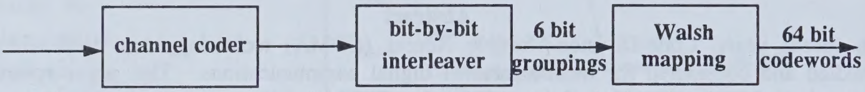


Figure 1a. Block diagram of the transmitter for the bit-by-bit interleaving scheme.

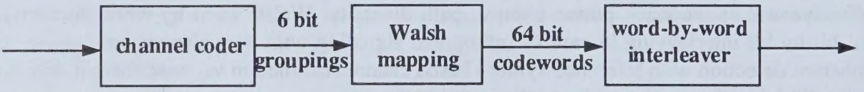


Figure 1b. Block diagram of the transmitter for the word-by-word interleaving scheme.

For the block interleaver under consideration, to minimize correlation among the signals, and to avoid the "wrap-around" effect [6], the best interleaver for a fixed block length $B=I\times J$ is to select J, the number of columns, to be slightly larger than the total number of coded symbols in a constraint length and $I = B/J$. Thus, for a constraint length 9, rate 1/3 convolutional coder, for the bit-by-bit interleaved case, $J=32$ (the total number of coded bits in a constraint length is 27) was chosen,(as described in TIA TR-45 [3]) For the word-by-word interleaved case, $J = 6$ (the constraint length is 5 coded Walsh symbols) was selected. For both techniques, the interleaved coded bits or Walsh codewords are finally spread with a PN sequence prior to transmission.

2.2. Base Station Receiver

The block diagram of the base station receiver with power control is shown in Figure 2. At the receiver, 64 Walsh correlators following the PN correlator are required to demodulate the received signals. The Walsh correlators output 64 coefficients (correlation values between each Walsh word and the received signal) for every 6 coded bits. The outputs of the corresponding Walsh correlators from each diversity branch are then square-law combined, and weighted with equal gain. The resulting 64 values are then used for power control and in the Viterbi decoder.

As for power control operation, since the decoder has large delay due to deinterleaving, information used for feedback power control must be extracted before deinterleaving and decoding. A detailed method for estimating the short term average SIR for power adjustments can be found in [7]. The basic concept is to take the ratio of the maximum of the output of Walsh correlators to the sum of the remaining values and average it over the power control updating period. The averaged value can be used to estimate the SIR, and it is compared with a given threshold for the decision to increment or decrement the power of the portable's transmitter. The power control information is uncoded (because of the interleaving delay) and is then sent to the user terminal through downlink transmission. The user terminal adjusts its transmitting power according to the received power control information.

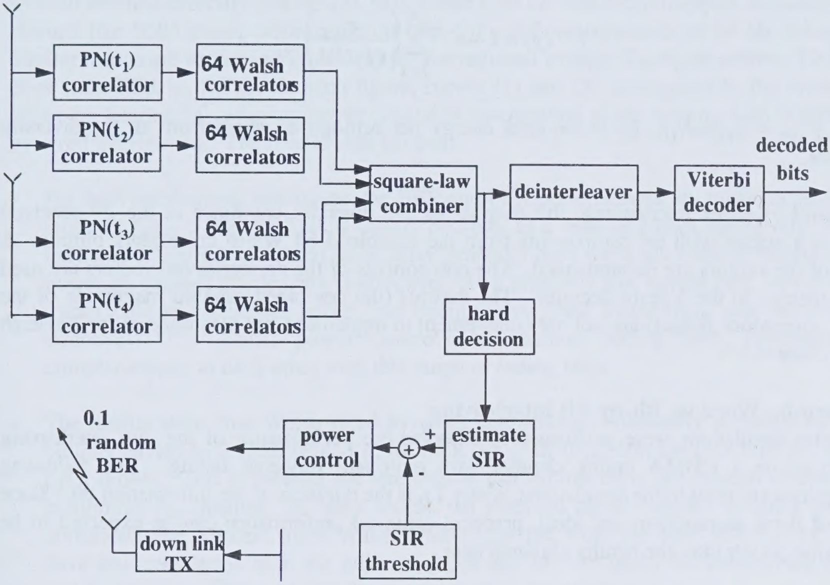

PN(t_1) correlator
64 Walsh correlators
PN(t_2) correlator
64 Walsh correlators
PN(t_3) correlator
64 Walsh correlators
PN(t_4) correlator
64 Walsh correlators
square-law combiner
deinterleaver
Viterbi decoder
decoded bits
hard decision
estimate SIR
+
-
power control
SIR threshold
0.1 random BER
down link TX

Figure 2. Block diagram of the basestation receiver.

The outputs of the Walsh correlators are handled differently for Viterbi decoding purposes. For bit-by-bit interleaving, the largest value r_{max} is obtained. The index of r_{max} consists of 6 bits which are the estimated received bits. A group of the received bits is then deinterleaved. The deinterleaved bits together with the log-likelihood function $L(r_{max})$ of r_{max}[8]

$$L(r_{\max}) = \ln\left(\frac{31}{32} + \frac{1}{32} \times \frac{f_c(r_{\max})F_i(r_{\max})}{f_i(r_{\max})F_c(r_{\max})}\right) \qquad \text{Eq.(1)}$$

are then used as symbols and weights in the Viterbi decoding algorithm. Where in Eq. (1), f_c and f_i are the probability density functions of the combined Walsh correlators' outputs with respect to the transmitted symbol and other (orthogonal) symbols; F_c, and F_i are the corresponding cumulative probability distribution functions. For the Rayleigh fading channel with m-branch diversity, the probability density functions are Nakagami and [8,9]

$$f_c^m(y) = \frac{y^{m-1} e^{-y/(\gamma_m+1)}}{(m-1)!(\gamma_m+1)^m}$$

$$f_i^m(y) = \frac{y^{m-1} e^{-y}}{(m-1)!}$$

$$F_c^m(y) = 1 - e^{-y/(1+\gamma_m)} \sum_{j=1}^{m} \frac{1}{(j-1)!}\left(\frac{y}{1+\gamma_m}\right)^{j-1}$$

$$F_i^m(y) = 1 - e^{-y} \sum_{j=1}^{m} \left(\frac{y}{(j-1)!}\right)$$

where $\gamma_m = E_c/(mN_0)$, E_c is the total energy per orthogonal signal from all the diversity branches.

For word-by-word interleaving, the outputs of the diversity combiner at the i^{th} received signal is a vector with 64 components from the combined 64 Walsh correlators output. A group of the vectors are deinterleaved. The components of the deinterleaved vectors are used as the metrics in the Viterbi decoder. The metrics (the combined squared magnitude of the Walsh correlators' output) are not only convenient to implement, but also optimal for Rayleigh fading channel.[10]

3. Word-by-Word vs. Bit-by-Bit Interleaving

Computer simulations were performed to examine the performance of the two interleaving techniques on a CDMA uplink channel with multipath Rayleigh fading. The following assumptions are used in the simulations, where T_b is the duration of the information bit. Since many of these assumptions are ideal, practical achieved performance can be expected to be somewhat poorer than the results obtained here.

1. For simplicity, spreading the bits by means of a higher chip rate is not considered.
2. Perfect symbol timing and carrier recovery are assumed.
3. Power estimation and control are updated every $T_p = 12\ T_b$ time periods.

4 1 dB power control step size is used.

5 Power control feedback delay (the delay from when the power control information is sent to when the power is adjusted) is T_p or $2T_p$; for convenience, we denote the feedback delay by $d = T_p$ or $d = 2\ T_p$.

6 For two branch antenna diversity, each branch has the same average signal strength and experiences independent Rayleigh fading. One correlator per branch is used.

7 10% random error on the power control command bit

8 192 information bits per frame, thus the overall interleaving delay is $192\ T_b$. The interleaver for bit-by-bit interleaving is 18× 32; for Walsh word-by-word interleaving, it is 16×6.

9 Interference is noise-like and has constant power level. Thus, signal-to-interference ratio and signal-to-noise ratio are treated the same in this paper.

10 The received signal was not quantized.

The radio channel encounters (1)time-varying multipath fading with a Rayleigh distributed fading envelope, (2)interference which is approximated as additive white Gaussian noise[11], and (3)multipath dispersion. The Rayleigh distributed multipath fading envelope can be generated by using a technique described by Jakes[12]. This technique assumes that the arrival angles of the multipath signals at the radio receiver are uniformly distributed over [0,2π], implying that the fading power spectrum is symmetric.

Figure 3(a) shows the required average signal-to-noise ratio E_b/N_0 to achieve 10^{-3} bit error ratio BER vs. f_dT_b for feedback delay $d=T_p$, with antenna diversity (curves (1), (2)) or without antenna diversity (curves (3), (4)), where f_d is the maximum Doppler frequency of the channel (for 9600 kb/sec information bit rate, $f_dT_b = .01$ corresponds to 96 Hz fading rate). Similar results are shown in Figure 3(b) for the required average E_b/N_0 to achieve 10^{-2} frame error ratio FER vs. f_dT_b. In each figure, curves (1) and (3) correspond to the system with bit-by-bit interleaving whereas curves (2) and (4) correspond to the system with Walsh word-by-word interleaving. The results indicate that:

- For both interleaving techniques, the BER and FER performances are quite stable over the range of $f_dT_b > .0025$. This is mainly because, at a relatively high fading rate, the feed back power control with fixed finite delay generally does not track channel variations well, however, channel interleaving is effective. At a relatively low fading rate, power control tracks channel variations better, but coding with fixed interleaving size becomes less effective[11] . Thus, power control and channel coding with interleaving are complementary to each other over this range of fading rates.
- The results show that Walsh word-by-word interleaving consistently outperforms bit-by-bit interleaving by 1.5 dB (Fig. 3(a) for BER) or 1 dB (Fig. 3(b) for FER) for the range of f_dT_b shown. This is because the interleaving size employed is not enough to completely randomize the channel. In other words, the received signal samples within a block are correlated, the adjacent input Walsh symbols for the word-by-word interleaving scheme have less correlation than the adjacent input bits for the bit-by-bit interleaving scheme, yielding better performance for the former than for the latter.

To investigate the effect of power control feedback delay on the performance, we have also performed simulations with feedback delay $d=2T_p$. Figures 4(a) and 4(b) show the effect of two power control feedback delays on the required average E_b/N_0 for 2-branch diversity. Figure 4(a) is for BER performance and 4(b) is for FER performance. It can be observed that the relative performance of word-by-word interleaving vs. bit-by-bit interleaving stays the same. However, the required E_b/N_0 increases as the feedback delay increases from $d=T_p$ to $d=2T_p$. This increase in E_b/N_0 is about 0.5 dB for $f_dT_b > 0.0025$ for both interleaving techniques. For relatively slow fading ($f_dT_b < 0.00125$) rates, the increase in E_b/N_0 is less sensitive to the feedback power control delay. This is because at this slow fading rate, the signal variation over some long period of time (relative to the feedback power control delay) is small, thus power control is more nearly ideal.

In table I and II, we summarize the above results for $d=T_p$ (which corresponds to 1.25 ms delay for 9.6 kbps user transmission rate), $2T_p$ and 2-branch antenna diversity for the fading channel. Notice that for the AWGN channel the results are obtained from the one-ray AWGN channel. The corresponding user speed for a given f_dT_b is also shown for a 9.6 kbps user transmission rate and 900 MHz carrier frequency..

In practical CDMA uplink communication channels, word-by-word interleaving outperforms bit-by-bit interleaving by more than 1.5 dB (1dB) in BER (FER). This difference in required E_b/N_0 is consistent over the entire range of vehicle speed and for single antenna or multiple antennas with diversity combining. As for implementation complexity, the computation requirement for word-by-word interleaving is slightly less, because the transition metric is directly obtained from the envelope detectors without extra computation. One the other hand, it requires about 25 to 30 times more memory than that required by bit-by-bit interleaving due to its larger interleaving buffer size.

Table I. The required average E_b/N_0 to achieve 10^{-3} BER for $d=T_p$ and $d=2T_p$

$d=T_p$	required average E_b/N_0 (dB), BER=10^{-3}				
	.01	.005	.0025	.00125 (f_dT_b)	AWGN
mph	72	36	18	9	-
w-w	3.82	3.81	3.71	2.90	3.9
b-b	5.47	5.28	5.53	4.37	5.4
$d=2T_p$	required average E_b/N_0 (dB), BER=10^{-3}				
w-w	4.47	4.45	4.19	3.08	3.9
b-b	6.07	6.14	5.95	4.67	5.4

Table II. The required average E_b/N_0 to achieve 10^{-2} FER for $d=T_p$ and $d=2T_p$

$d=T_p$	required average E_b/N_0 (dB), FER=10^{-2}				
	.01	.005	.0025	.00125 (f_dT_b)	AWGN
mph	72	36	18	9	-
w-w	4.23	4.15	3.92	3.15	4.5
b-b	5.16	5.01	5.0	4.12	5.6
$d=2T_p$	required average E_b/N_0 (dB), FER=10^{-2}				
w-w	4.87	4.7	4.36	3.36	4.5
b-b	5.73	5.81	5.42	4.22	5.6

For the bit-by-bit interleaving scheme, Eq. (1) is used to calculate the metric for Viterbi decoding in this paper. A more complicated metric which was derived from the log-likelihood functions of all correlator outputs has been investigated.[13] It is shown [13] that using the same bit-by-bit interleaving scheme, performance improvement of 1 to 1.2 dB is achieved through the use of more sophisticated metrics.

4. Performance of multiple correlators per antenna

In this section, the effect of the use of multiple correlators per antenna in the various multipath delay-spread environments is shown. The purpose of two correlators per antenna branch is to capture the two strongest signal paths to minimize the effect of multipath dispersion.

Results were obtained through computer simulations. Word-by-word interleaving is used. Simulations assumptions are listed in the previous section except that power control delay of 2 units (i.e. $d=2T_p$) is employed. We also assumed that all of the signal power from each antenna has been fully captured by the correlator (for the case of a single correlator per antenna) or the two correlators (for the case of two correlators per antenna) and each delay path experiences independent Rayleigh fading. Three pairs of power distributions between the two delay paths are considered. They are: (0.5, 0.5), (0.9, 0.1) and (0.99, 0.01). The power distribution pair (a_1, a_2) represents that one of the correlators captures a_1 and the other captures a_2 of total average signal power, assuming that total signal power is normalized to 1.

In Figure 5, the required average E_b/N_0 to achieve 10^{-3} decoded BER is shown for feedback power control delay of 2 units with 1 dB power control step size. Results of the channel using one correlator that captures all the power(curve (a)) or two correlators with various power distributions (curves (b), (c), (d)) are shown. The required E_b/N_0 for a channel using two correlators per antenna with power distribution of (0.99, 0.01) (curve (d)) is more than that for a channel using one correlator per antenna that captures all the power (curve (a)) over the entire range of the channel fading rate. The difference can be more than 1 dB. This indicates that when the signal strength difference between the two delay paths is 20 dB or more, the system should turn off one of the correlators or have the two correlators both locked onto the strongest path. The performance difference among the unequal power distribution paths are mainly due to the equal-gain combining.(It is known that if all the average path strengths are known and all path strengths are independent, the optimal combining scheme is to weight each path with the ratio of the signal to noise strength [14].) Notice that at 0 fading rate, 1 dB difference is obtained for all cases, this is caused by the noncoherent combining loss.

6. Discussions and Conclusions

In this paper, all the results are based on noncoherent detection in the CDMA uplink. A number of other research results in the literature are also based on this same assumption.. Recently, Ling[15] has proposed a coherent detection scheme using reference-symbol based channel estimation for CDMA uplink communications. The concept is to insert the reference-symbols periodically at the transmitter. At the receiver, since the reference-symbols are known, the receiving signals corresponding to the reference symbols can be extracted and used to generate channel estimation. It has been shown [15] that using this scheme with perfect timing recovery, 1 to 1.3 dB improvement over the noncoherent detection scheme with Walsh word interleaving is obtained. Although degradation due to symbol timing error is unknown, this approach provide an interesting alternative to the noncoherent detection scheme.

This paper has focused on the uplink performance of a CDMA system. We conclude that for noncoherent detection with Walsh orthogonal signals: (1)word-by-word interleaving is preferred over the bit-by-bit interleaving due to its superior performance; (2)for an environment with unequal power distribution in components of the delay spread profile and more than 20 dB power difference among the delay paths, the system should turn off one of its correlator to avoid capturing the unwanted noise which results in performance degradation.

Notice that all these simulations assume that the receiver's correlators capture 100% of the desired signal's energy. An analysis of typical urban delay spread profiles showed that about one dB energy loss can be expected on average for practical receivers in urban environments[16].

In this paper, we did not discuss other issues relative to CDMA systems such as soft hand-off, voice quality, uncaptured time-dispersed energy and other non-idealized factors. All of these factors may affect the capacity performance of a CDMA system.

REFERENCE

---

[1] K. S. Gilhousen, et. al.: "On the Capacity of a Cellular CDMA System", *IEEE Trans. on Veh. Tech.*, Vol. 40, No. 2, pp. 303-312, May 1991.

---

[2] Schilling, D. L., Pickholtz, R. L., and Milstein, L. B.: "Spread Spectrum Goes Commercial," *IEEE Spectrum*, August 1990, pp. 40-45.

[3] TIA TR45 Document PN-3118: "Mobile Station - Base Station Compatibility Standard for Dual-Mode Wideband Spread Spectrum Cellular System," Dec. 1992.

[4] L. F. Chang and N. R. Sollenberger, "Comparison of Two Interleaving Techniques for CDMA Radio Communications Systems," *Proc. of VTC'92*, pp. 275-278, May 1992, Denver, CO.

[5] F. Ling and D. Falconer, "Orthogonal/Convolutional Coding for Reverse Channel CDMA Communication," *Proc. of VTC'92*, pp. 63-66, May 1992, Denver, CO.

[6] F. Simpson and J. M. Holtzman, "CDMA Power Control, Interleaving and Coding," *Proc. VTC'91*, pp. 362-367, May, 1991.

[7] L. F. Chang, S. Ariyavisitakul, "Performance of a CDMA Radio Communications System with Feed-Back Power Control and Multipath Dispersion," *Globecom'91 conference record*, pp. 29.6.1-29.6.5, Phoenix, AZ., Dec. 2-5, 1991.

[8] A. J. Viterbi, A. M. Viterbi and E. Zehavi, "Performance of Power Controlled Wideband Terrestrial Digital Communication," *IEEE Trans. on Commun.*, vol. 41, No. 4, pp. 559-569, April 1993.

[9] W. E. Ryan and S. G. Wilson, "Two Classes of Convolutional Codes over GF(q) for q-ary Orthogonal Signaling," *IEEE trans. Commun.*, vol. COM-39, No. 1, Jan. 1991, pp. 30-40.

[10] L. F. Chang, F. Ling, D. D. Falconer and N. R. Sollenberger, "Comparison of Two Convolutional Orthogonal Coding Techniques for CDMA Radio Communications Systems," submitted for *IEEE Trans. on Commun.*

[11] D. C. Cox, "Universal Digital Portable Radio Communications," *IEEE Proceedings*, vol. 75, April 1987, pp. 436-477.

[12] W. C. Jakes, Jr., Ed., Microwave Mobile Communications, New York: Wiley, 1974.

[13] Y.-J. Liu, "Soft Decision Decoding of a Bit-Interleaved Convolutionally Encoded Code Division Multiple Access System over Rayleigh Fading," *Proc. of PIMRC'92*, pp. 128-132, Oct. 1992.

[14] G. L. Turin, "Introduction to Spread-Spectrum Antimultipath Techniques and Their Application to Urban Digital Radio," *IEEE Proceedings*, vol. 68, No. 3, pp. 328-353, March 1980.

[15] F. Ling, "Coherent Detection with Reference-Symbol Based Channel Estimation For Direct Sequence CDMA Uplink Communications," *Proc. of VTC'93*, pp. 400-403, May 1993.

[16] L. F. Chang, "Dispersive Fading Effect in CDMA Radio Systems," *IEEE ICUPC'92*, pp. 185-189, Oct. 1992.

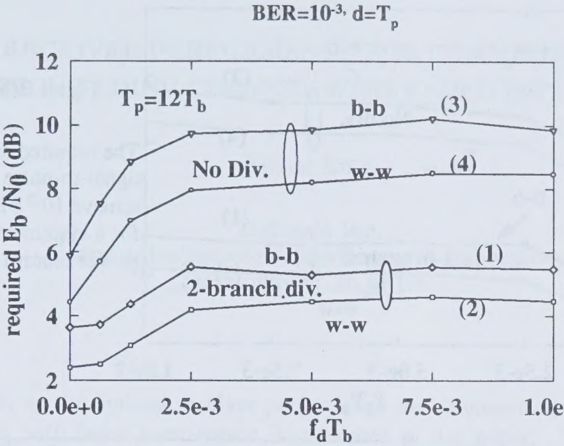


Figure 3(a). The required average signal-to-noise ratio to achieve 10^{-3} BER vs. f_dT_b for a system with feedback power control delay d=T_p.

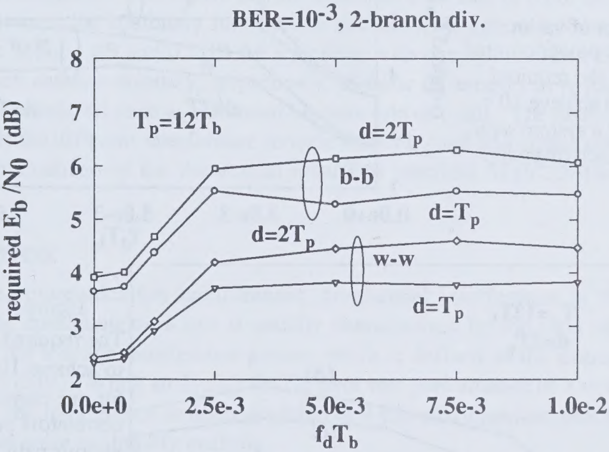


Figure 4(a). The effect of various feedback power control delay on the required E_b/N_0 to achieve10^{-3} BER for a system with 2-branch diversity.

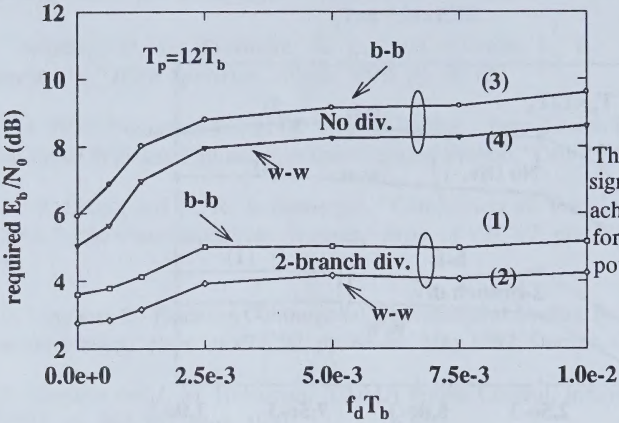


Figure 3(b)

The required average signal-to-noise ratio to achieve 10^{-2} FER vs. f_dT_b for a system with feedback power control delay $d=T_p$.

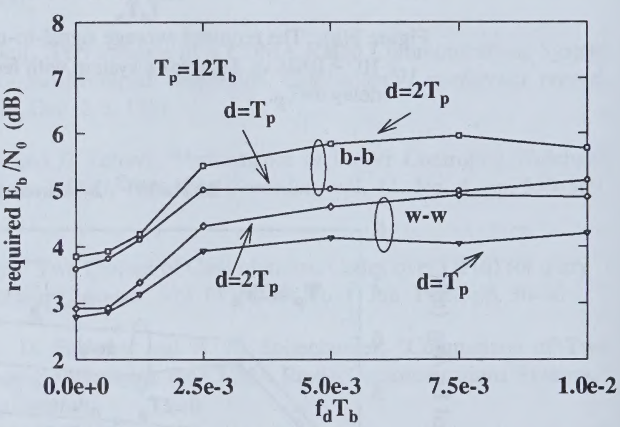


Figure 4(b)

The effect of various feedback power control delay on the required E_b/N_0 to achieve 10^{-2} FER for a system with two-branch diversity.

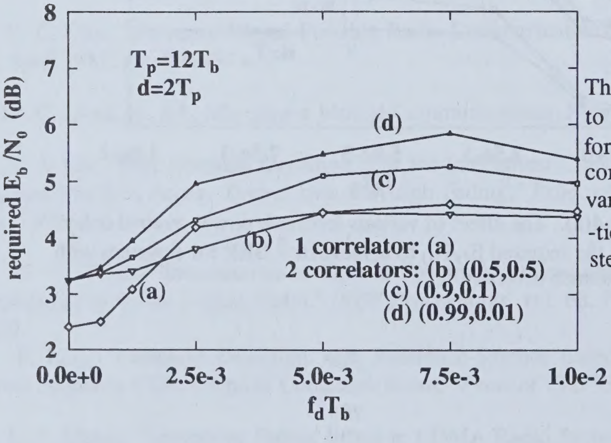


Figure 5

The required average E_b/N_0 to achieve 10^{-3} BER vs. f_dT_b for one correlator ((a)), two correlators per antenna with various path power distributions, 1 dB power control step size.

OPTIMAL RECEIVER PERFORMANCE FOR COMMUNICATION OVER A RAYLEIGH FADING CHANNEL WITH FADED INTERFERENCE

Fuyun Ling

Motorola Inc.
Communications Systems Research Laboratory
Schaumburg, IL 60196

ABSTRACT:

An analysis on the optimal receiver performance for communication over a Rayleigh fading channel with faded interference is presented in this paper. The interference is modeled as Gaussian noise with power being χ-square distributed with 2m degrees of freedom, and the channel is a Rayleigh flat fading channel. Under these assumptions, exact expressions of uncoded BER and upper bounds of BER for convolutional coding are derived for an optimal receiver with one or two receiving antennas. The cut-off rates (R_0) for these channels are also computed. It is shown that for a channel error rate of 0.08, the difference in required SIR between the stationary interference and the Rayleigh and Nakagami-2 faded interferences are about 1 dB and 0.5 dB for a receiver with one antenna, and 2.0 dB and 1.0 dB for two-branch antenna diversity, respectively. Similar differences in required SIR exist for a 32-state convolutional code at a decoded bit error rate of 0.001. The difference between the upper bounds for different interference models shows a good agreement with simulation results. The implications of the theoretical results to practical SFH/CDMA systems are discussed.

I. INTRODUCTION

In a mobile communication environment, co-channel interference is the dominant impairment. The communication link is usually characterized by the ratio of the average signal power to the average interference power, which is defined as the average signal-to-interference ratio (SIR). When analyzing the bit error rate performance of a receiver in such an environment, the interference is often modeled as a stationary random process or simply as white Gaussian noise to simplify analysis.

Since the co-channel interference is the desired signal to or from other users, it experiences fading during propagation. Further, in a slow frequency hopping/code division multiple access (SFH/CDMA) system, the frequency-hop codes may be selected such that the set of interfering subscribers changes for every frequency hop. While it may be valid to model the interference in a direct-sequence CDMA (DS/CDMA) system as stationary noise due to the averaging effect of the large number of interferers (especially for the uplink, i.e., from mobile to base), this is not a correct model for SFH/CDMA. Analysis of the performance of an SFH/CDMA system must take both of these variations into account.

The bit error rate (BER) performance is a function of the distribution of the SIR, which is a function of the signal power distribution and interference power distribution. As a consequence, the single parameter of average SIR cannot fully define the link performance. Although the effect of signal fading on the BER performance has been discussed thoroughly in the literature, the impact of interference fading is well understood. In this paper, we analyze the effect of signal and interference fading for a two-channel diversity receiver assuming that the demodulator has perfect knowledge of the SIR for every frequency hop.

For mobile cellular communication, the amplitude of the received signal can be modeled as a Rayleigh fading process. Similarly, the amplitude of the co-channel interference can be modeled as a Rayleigh, or a Nakagami fading process, when the interference is from more than one transmitter.

In the analysis, the magnitude and phase of the channel are assumed to be constants for each hop. The interference is modeled as a stationary Gaussian process during a single frequency-hop period; over a long time period the probability density of the interference power is χ-square with 2m degrees of freedom. In the sequel, we shall call such interference *Nakagami-m faded interference*. For m=1, this power density becomes Rayleigh. The receiver is assumed to be coherent and optimal. Specifically, the receiver/demodulator includes an ideal matched filter and the SIR at its output is known or can be measured exactly. Maximal ratio diversity combining is used and maximum likelihood metrics are used for soft-decision convolutional decoding. The channel is assumed to be memoryless, i. e., the samples used in combining and decoding are independent. While these assumptions are idealized, they can be approximately realized in a slow-frequency-hopping (SFH) communication system.

In Section II, the probability density functions (pdf) of the SIR distribution of the signal at the output of the matched filter with and without two-channel diversity are derived. Exact expressions of BER for uncoded BPSK/QPSK signaling are then computed. In Section III, the coded performance is evaluated by computing union-Bhattacharyya upper bounds [1] for specific convolutional codes and by computing the computational cut-off rate R_0 for the assumed channels. Section IV shows the cumulative distribution of the SIR, which provides insight into the obtained results on the receiver performance. Section V discusses the implications of the theoretical results to practical SFH/CDMA systems and concludes the paper.

II. UNCODED BER FOR BPSK/QPSK SIGNALING OVER RAYLEIGH FADING CHANNELS WITH FADED INTERFERENCE.

Given the assumptions described above, the sampled signal x at the output of the demodulator can be expressed as

$$x = \alpha U + n, \qquad (1)$$

where α is the channel gain, U is the sampled matched filter output due to the desired signal, and n is the interference portion of the demodulator output. The probability density function of n is Gaussian with zero mean and variance S.

The bit error probability P_b of BPSK/QPSK signaling over a stationary additive white Gaussian noise (AWGN) channel can be expressed as [2]

$$P_b(\gamma_b) = \frac{1}{2}\mathrm{erfc}(\sqrt{\gamma_b}), \tag{2}$$

where $\gamma_b = E/S$ is the SIR per bit at the demodulator output, in which E is equal to $\alpha^2|U|^2$ for BPSK signaling and $\alpha^2|U|^2/2$ for QPSK signaling. For a Rayleigh fading channel with Nakagami-m faded interference, E is χ-square distributed with 2 degrees of freedom, and S is χ-square distributed with 2m degrees of freedom. Let $\bar{\gamma}_b$ denote the average SIR per bit. Without loss of generality, the average interference power and signal power are assumed to be 1 and $\bar{\gamma}_b$, respectively. Then, the probability density functions (pdf) of E and S are [2]

$$f_E(E,\bar{\gamma}_b) = \frac{1}{\bar{\gamma}_b}e^{-E/\bar{\gamma}_b} \tag{3}$$

and

$$f_S^{(m)}(S) = \frac{m^m}{(m-1)!}S^{m-1}e^{-mS}, \text{ m = 1, 2, ... }, \tag{4}$$

respectively.

The random variable $\gamma_b = E/S$ is the quotient of two random variables E and S. The pdf of γ_b can be computed using [3],

$$f_{\gamma_b}^{(m)}(\gamma_b,\bar{\gamma}_b) = \int_0^\infty y f_E(y\gamma_b,\bar{\gamma}_b) f_S^{(m)}(y)dy. \tag{5}$$

By substituting (3) and (4) into (5) and evaluating the integral, we obtain

$$f_{\gamma_b}^{(m)}(\gamma_b,\bar{\gamma}_b) = \frac{1}{\bar{\gamma}_b(1+\gamma_b/m\bar{\gamma}_b)^{m+1}}. \tag{6}$$

The average bit error probability for the average SIR of $\bar{\gamma}_b$ is

$$P_b^{(m)}(\bar{\gamma}_b) = \frac{1}{2}\int_0^\infty \mathrm{erfc}(\sqrt{\gamma_b}) f_{\gamma_b}^{(m)}(\gamma_b,\bar{\gamma}_b)d\gamma_b, \tag{7}$$

which can be evaluated numerically.

Two-Branch Antenna Diversity

The received signals x_i at the output of the demodulator connected to the i-th antenna, i = 1, 2, is

$$x_i = \alpha_i U + n_i, \tag{8}$$

where α_i is the i-th Rayleigh distributed channel gain, and n_i represents the interference, which is a zero mean Gaussian random variable with variance S_i. The combining coefficients for the maximal ratio combiner are α_1/S_1 and α_2/S_2. Thus, the output of the combiner is

$$z = \frac{\alpha_1^2}{S_1}U + \frac{\alpha_1}{S_1}n_1 + \frac{\alpha_2^2}{S_2}U + \frac{\alpha_2}{S_2}n_2 = \left(\frac{\alpha_1^2}{S_1} + \frac{\alpha_2^2}{S_2}\right)U + \frac{\alpha_1}{S_1}n_1 + \frac{\alpha_2}{S_2}n_2. \tag{9}$$

The signal-to-interference ratio at the combiner output is

$$\begin{aligned}\gamma_{b,z} &= \left(\frac{\alpha_1^2}{S_1} + \frac{\alpha_2^2}{S_2}\right)^2 |U|^2 \Bigg/ \left(\left(\frac{\alpha_1}{S_1}\right)^2 E[|n_1|^2] + \left(\frac{\alpha_2}{S_2}\right)^2 E[|n_2|^2]\right) \\ &= \left(\frac{\alpha_1^2}{S_1} + \frac{\alpha_2^2}{S_2}\right)^2 |U|^2 \Bigg/ \left(\left(\frac{\alpha_1}{S_1}\right)^2 S_1 + \left(\frac{\alpha_2}{S_2}\right)^2 S_2\right) = \left(\frac{\alpha_1^2}{S_1} + \frac{\alpha_2^2}{S_2}\right)|U|^2 = \frac{E_1}{S_1} + \frac{E_2}{S_2} = \gamma_{b,1} + \gamma_{b,2},\end{aligned} \tag{10}$$

where $\gamma_{b,i}$ is the SIR for diversity branch i. Under the assumptions stated above, $\gamma_{b,1}$ and $\gamma_{b,2}$ are independent and have the pdf given by (6). The pdf of $\gamma_{b,z}$ is the convolution of (6) with itself. Evaluating the convolution for m = 1 and 2 yields

$$f^{(1)}_{\gamma_{b,z}}(\gamma_b, \bar{\gamma}_b) = \frac{2\gamma_b}{\bar{\gamma}_b^2(1+\gamma_b/\bar{\gamma}_b)(2+\gamma_b/\bar{\gamma}_b)^2} + \frac{4\ln(1+\gamma_b/\bar{\gamma}_b)}{\bar{\gamma}_b(2+\gamma_b/\bar{\gamma}_b)^3}, \text{ m = 1}, \tag{11}$$

and

$$f^{(2)}_{\gamma_{b,z}}(\gamma_b, \bar{\gamma}_b) = \frac{\gamma_b[6+(2+\gamma_b/2\bar{\gamma}_b)^2]}{\bar{\gamma}_b^2(1+\gamma_b/2\bar{\gamma}_b)(2+\gamma_b/2\bar{\gamma}_b)^4} + \frac{24\ln(1+\gamma_b/2\bar{\gamma}_b)}{\bar{\gamma}_b(2+\gamma_b/2\bar{\gamma}_b)^5}, \text{ m = 2}. \tag{12}$$

For m > 2, the pdf of the SIR can be evaluated similarly with more complex expressions. The uncoded BER can be computed by using (11) and (12) in place of $f^{(m)}_{\gamma_b}$ in (7).

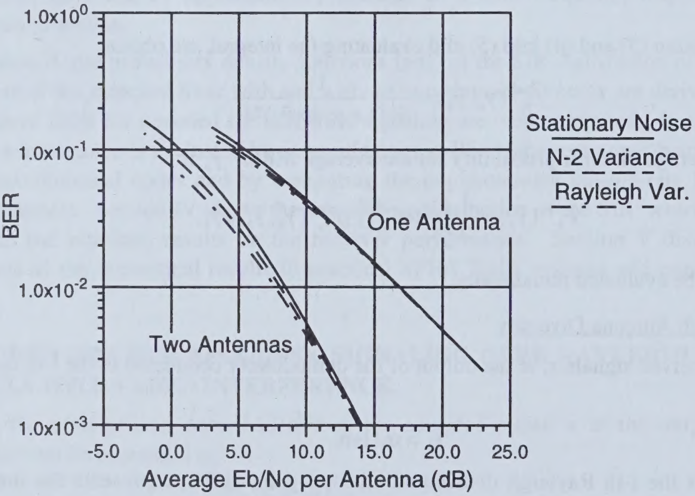


Figure 1. Uncoded BER performance for stationary and faded noise

Figure 1 depicts the computed uncoded (channel) BER for one and two-branch antenna diversity combining, and for Rayleigh fading (m=1) and Nakagami-2 (N-2, m=2) faded interference models. For comparison the BER was also calculated for stationary interference;

this result is also depicted in Figure 1. Figure 1 shows that, at the same average SIR, the BERs for faded interference are always less than that for stationary interference. The larger the value of m is, the smaller the difference becomes. It is also noteworthy that the difference in error rate diminishes when SIR increases. However, for the channel error rate of interest, i.e., for a BER of 0.08, corresponding to the decoded error rate of about 0.001 for a convolutional code described later, the differences in SIR between the stationary interference and faded interference with m = 1 and 2 are about 1 dB and 0.5 dB for a receiver with one antenna, and 2.0 dB and 1.0 dB for two-branch antenna diversity combining, respectively.

III. COMPUTATIONAL CUT-OFF RATE AND CODED PERFORMANCE

We have derived the pdfs of SIRs with no-diversity combining and two-branch antenna diversity combining. These expressions were used to compute the uncoded BER. Unfortunately, it is not possible to obtain pdfs in closed form for three branch diversity combining and above. As a result, it is not easy to evaluate the union bound of decoded BER based on the exact pairwise error probability P_2. In what follows coded performance is evaluated through the looser union-Bhattacharyya bound [1] and the computational cut-off rate R_0.

For a particular convolutional code, the bit error rate is upper bounded by the union-Bhattacharyya bound [1,6]

$$P_b < \frac{1}{2}\sum_{d=d_f}^{\infty}\beta_d Z^d, \tag{13}$$

where β_l is the total number of information bits for all the convolutional code trellis paths that have path weight equal to l and, for a stationary binary-input stationary AWGN channel, Z is

$$Z(\gamma_c) = e^{-\gamma_c}, \tag{14}$$

where γ_c is the SIR per (channel) bit at the demodulator or diversity combiner output. Note that β_l is a function of the specific convolutional code being considered. For the Rayleigh fading channel with Rayleigh and Nakagami-m faded interference, the performance parameter Z is the average of (14) over the SIR pdfs given by (6), (11) or (12). That is

$$Z_f(\bar{\gamma}_c) = \int_0^{\infty} e^{-\gamma_c} p_f(\gamma_c, \bar{\gamma}_c) d\gamma_c. \tag{15}$$

This analysis requires the channel to be memoryless. The required independence is achieved using interleaving. Using (13) through (15) the bit error rate may be calculated numerically.

Figure 2 depicts the upper bound of Eq. (13) for a 32-state convolutional code with code generator 75 and 53 (in octal) used in the proposed SFH PCS system described in [5]. Bit error rate results calculated for stationary interference are also illustrated in Figure 2. For two-antenna diversity, the SIR used in Figure 2 is the SIR *per antenna.* It can be seen from the figure that, at a BER of 10^{-3}, the difference in performance between the stationary and Rayleigh faded interferences is about 1.2 dB for the case of one antenna, and about 2.5 dB for two-branch antenna diversity combining. Thus, taking the fading of the interference into account is necessary for accurate performance estimates. Simulation results of an ideal receiver for a BER = 0.001 are also plotted. At a BER of 10^{-3}, the upper bounds differ from the simulation results by between 0.9 dB and 1.1 dB. It is important to note that the difference between the upper bounds for different interference models is consistent with the simulation.

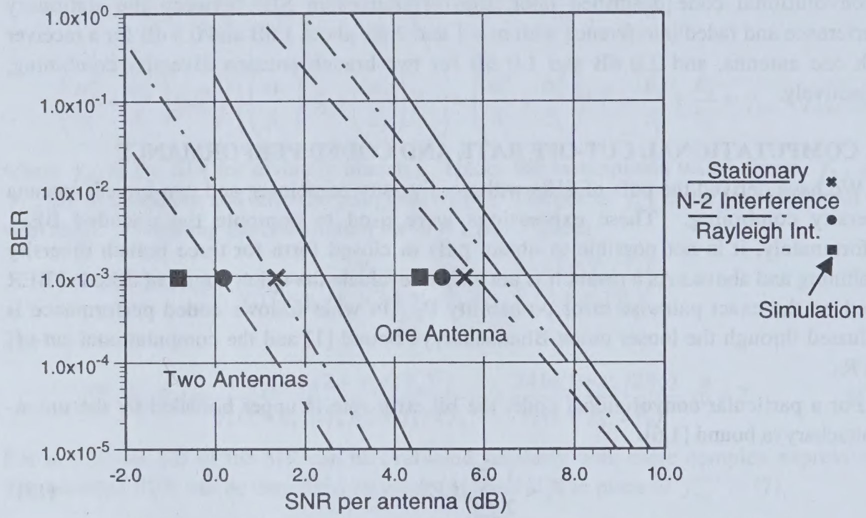


Figure 2. Upper bound and simulation results of decoded BER for 32-state convolutional code over Rayleigh fading channel with faded and stationary noise.

The computational cut-off rate R_0 is related to Z by

$$R_0(\bar{\gamma}_c) = 1 - \log_2\left[1 + Z_f(\bar{\gamma}_c)\right]. \qquad (16)$$

Equation (16) expresses the cut-off rate as a function of average channel SIR. It is more informative to express it in terms of the SIR per information bit. Assuming that coding is applied at the cut-off rate, the channel SIR γ_c is related to the SIR per information bit γ_b by

$$\bar{\gamma}_b \equiv \bar{\gamma}_c / R_0(\bar{\gamma}_c). \qquad (17)$$

Figures 3 and 4 show the relationship between R_0 and $\bar{\gamma}_b$ for no diversity and two-branch antenna diversity, respectively. In both figures, three different interference conditions are shown: 1) Rayleigh faded interference, 2) Nakagami-m faded interference, and 3) stationary non-faded interference. For these figures, SIR is the ratio of the total average signal to average interference for the case of two-branch antenna diversity. The SIR values will be 3 dB lower for two-branch antenna diversity combining if we consider the SIR per antenna. These figures illustrate the SIR required to achieve error free communications at a code rate equal to R_0. It is noteworthy that the lower the coding rate, the larger the difference in SIR between the faded and stationary interference. Furthermore, it seems from the figures that the required SIR continuously decreases as the coding rate goes to zero for Rayleigh faded interference. This may imply that the capacity of the Rayleigh fading channel with Rayleigh faded interference is unlimited when there is no additive noise. Although this observation is of theoretical interest, it is unrealistic, because there always exists additive noise. It should also be noted that the lower the coding rate is, the more difficult the realization of an optimal receiver becomes, because it is more difficult to accurately estimate the low channel SIR at a low coding rate.

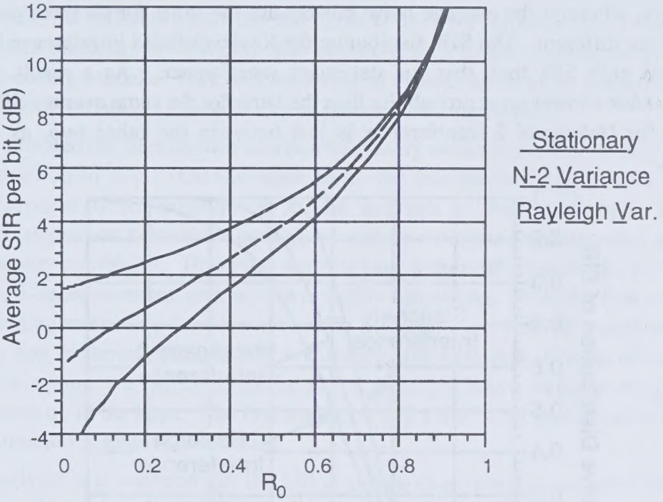


Figure 3. R_0 vs. Required Average SIR (no diversity).

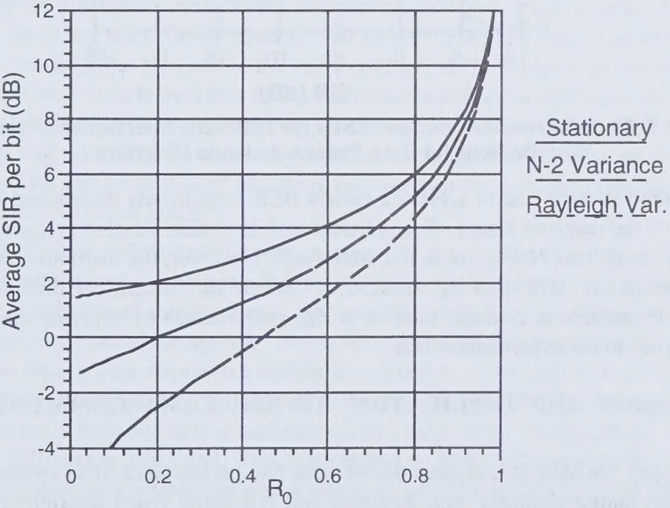


Figure 4. R_0 vs. Required Average SIR (two-branch antenna diversity).

IV. CUMULATIVE DISTRIBUTION OF SIR

The difference in performances for different types of interference statistics can be better understood from the cumulative probability distributions (cdf) of SIRs depicted in Figure 5. In the figure, we plot the cdfs of SIRs for the three types of interference with the same average noise powers for the case of two-branch antenna diversity. The average signal power and the average noise power are set to be equal, i. e., at 0 dB average SIR. It can be seen from the figure that, although the average noise powers are the same for all three case, the SIR distributions are different. The SIR distribution for Rayleigh faded interference has a larger percentage of high SIR than that for stationary interference. As a result, the formal distribution yields a lower error probability than the latter for the same average SIR. The SIR distribution for Nakagami-2 interference is just between the other two, as is its BER performance.

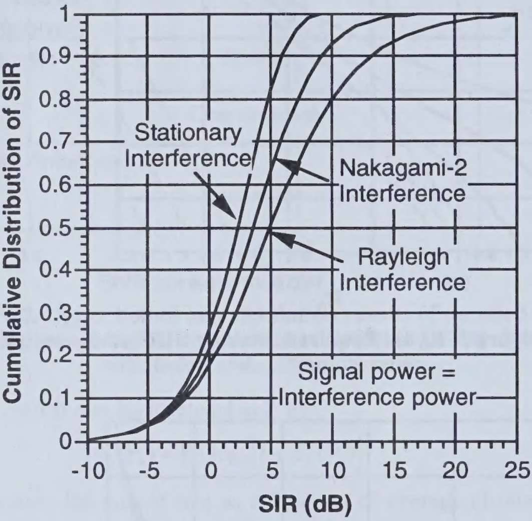


Figure 5. Cumulative Distribution of SIR for Different Interference Variance Distributions for Two Branch Antenna Diversity

The required average SIR to achieve a certain BER is ultimately determined by the SIR distribution of the received signal. For practical mobile communication systems, the noise distributions are neither Nakagami-m nor stationary. However, the distribution of the SIR can be conveniently evaluated by simulation. When such empirical distributions are available, it is possible to evaluate their error rate performance by applying the analytical tools used above to the experimental data.

V. DISCUSSION AND IMPLICATION TO SFH CDMA COMMUNICATION SYSTEMS

In this paper, we have investigated the bit error rate performance of an optimal receiver over Rayleigh fading channels with Rayleigh and Nakagami faded interference. It was shown that an optimal receiver requires less average SIR when the interference is Rayleigh or

Nakagami faded than that for stationary interference. Thus, proper accounting for the interference fading statistics is important to achieve reliable performance estimates.

We computed the uncoded BER for Rayleigh fading channels with Rayleigh or Nakagami faded interference by first obtaining the exact expressions of the SIR distribution. The decoded BER for a specific convolutional code was evaluated using the union-Bhattacharyya upper bound and the bound was then compared with simulation results. It is noteworthy that the difference between the upper bounds for different interference models is consistent with simulation. We may also compute such upper bounds for more complex interference fading models. This will be useful in evaluating receiver performance in such channels if the distribution of the SIR can be determined empirically.

We have treated the interference as a non-stationary random process, namely, its variance changes from hop to hop. On the other hand, the interference can also be viewed as a stationary process if we consider it as the average of the interference in all hops. Performance evaluation from these two views will result in the same uncoded average BER without diversity combining. However, we will have rather different results for the decoded BER with soft-decision metric and/or with diversity combining. From the first viewpoint, the optimal decoding metric is a function of the hop SIR and, thus, must be computed differently from hop to hop, assuming that the SIR is estimated for each hop. On the other hand, from the second viewpoint, we will compute the decoding metric based on the average statistics of the interference of all the hops. The first approach will yield better performance provided the hop SIR estimation is relatively accurate.

In the analysis, it is assumed that the SIR is known exactly so that maximal ratio diversity combining and maximum likelihood decoding metrics can be used in the receiver. Practically, the SIR can only be estimated with a certain accuracy. As a result, some implementation loss will be unavoidable. However, in an SFH/CDMA system with a TDM structure [5] the SIR within one slot does not vary much. Thus, it can be estimated relatively accurately by using the training sequence to minimize such a loss. Another idealized assumption made in the analysis is the independence of the samples at the output of the receiver front end. It is known that this is realizable if sufficient interleaving is employed. Practically, interleaving depth is limited by the allowable communication delay. As a result, degradation will occur if the interleaving depth is not large enough. The performance of a realistic receiver can be evaluated by taking into account the implementational losses with reference to the analytical result.

In a SFH/CDMA cellular mobile radio system, the distribution of interference power for each hop is determined by both the Rayleigh fading of the propagation channel and the geographic distribution of the interferers. For example, consider the base-to-mobile path in a power controlled SFH/CDMA system. The power control mechanism adjusts the transmitted signal power from a base station to a mobile as a function of the path loss between the base and the mobile. This path loss varies from mobile to mobile. The interference to a mobile consists of signals from one or two base stations in nearby cells. For each hop, these signals will be for different mobiles and, thus, transmitted at different power levels. Due to the variation in transmitted interfering signal power in addition to the Rayleigh fading of the propagation channel, the received interference power varies from frequency hop to frequency hop. Due to interleaving this hop-to-hop variation becomes a bit-to-bit variation. We have observed that bit error rate performance is better when the fading of the interference is properly taken into account. Thus we have also demonstrated that a system in which the interference varies from bit-to-bit, for example SFH/CDMA, will, for the same average SIR, perform better than a system in which the interference is stationary. Noting, that the interference for a direct-sequence (DS) CDMA system is approximately stationary, we conjecture that, within the limits of this approximate analysis, a SFH/CDMA system will likely outperform a DS/CDMA system.

ACKNOWLEDGMENT

The author wishes to thank Dr. David Borth for his support and encouragement, and to thank Dr. Roger Peterson and Dr. Joseph Nowack for their helpful comments and stimulating discussions.

REFERENCES

[1] A. J. Viterbi and J. K. Omura, *Principles of Digital Communications and Coding*, McGraw-Hill, NY, 1979.

[2] J. G. Proakis, *Digital Communications*, McGraw-Hill, New York, 1983.

[3] A. Papoulis, *Probability, Random Variables, and Stochastic Process*, McGraw-Hill, New York, 1965.

[4] A. J. Viterbi, E. Zehavi and A. M. Viterbi, "Performance of Power Controlled Wideband Terrestrial Digital Communication," *IEEE Transactions on Communications*, Vol. 41, pp. 559 - 569, April 1993.

[5] P. D. Rasky, G. M. Chiasson, and D. E. Borth, "An Experimental Slow Frequency Hopped Personal Communication System for the Proposed U. S. 1850-1990 MHz Band," *Proc. of 2nd International Conference on Universal Personal Communications*, Ottawa, Canada, October 1993.

[6] A. J. Viterbi, "Convolutional Codes and Their Performance in Communications Systems," *IEEE Transactions on Communications Technology*, October 1971.

ADVANCED CDMA BASED ON CO-CHANNEL INTERFERENCE CANCELLATION

Ryuji KOHNO

Division of Electrical and Computer Engineering
Faculty of Engineering
Yokohama National University
156 Tokiwadai, Hodogaya-ku, Yokohama JAPAN 240
Phone: +81-45-338-1177, Fax: +81-45-338-1157
E-mail: `kohno@kohnolab.dnj.ynu.ac.jp`

Abstract

This paper proposes an advanced CDMA (code division multiple access) based on co-channel interference cancellation (CCI), i. e. A-CDMA through a review of the challenges and opportunities that arise for improving user capacity of spread spectrum multiple access (SSMA) or CDMA based on spread spectrum techniques. CCI in SSMA is the most dominant factor in the limitation of user capacity, while capacities of FDMA and TDMA are primarily bandwidth limited. From an information theoretical viewpoint, however, an optimum multiuser receiver for SSMA can be derived by utilizing CCI as redundant information distributed among multiple accessing users. In practical channels such as personal, indoor, mobile radio communication channels, more feasible receivers are required to combat with multipath fading that causes estimation errors of channel characteristics due to time variance. An overview of several CCI cancellation schemes to carry out sub-optimum but practical multi-user receivers for SSMA is provided through discussion of optimum receiver. In particular, these schemes are introduced mainly from our works by classifying them to the processing on temporal, spatial and code domains for direct-sequence (DS) SSMA and frequency-hopping (FH) one.

1. Introduction

The Japanese standard for wireless local area networks (LANs) that was established December 1992 is accelerating research and development of spread spectrum (SS) systems for consumer and personal communications. Its main regulation is to use SS modulation such as direct-sequence (DS), frequency-hopping (FH) or their hybrid in 2.4 GHz band. FCC in USA opened the industrial, scientific, and medical (ISM) frequency band for unlicensed operation of SS systems in 1985 and modified its rules to permit wider bandwidth in 1990.

Moreover, SS techniques have a shot at becoming a mass-market technology in the digital standard for cellular telephones. Both CDMA and TDMA have been proposed as the transmission standard for digital cellular telephone systems and products. Their battle is ongoing. CDMA offers up to about four to six times more capacity than first generation TDMA. That improvement of capacity depends on some additional techniques such as power control, voice activation, soft hand-off, and sectorization.

Efficient utilization of available frequency spectra is of major importance in not only wireless LANs and digital cellular telephones but also other consumer and personal communications. If mature techniques to improve capacity of CDMA more are established, CDMA based on SS techniques or SSMA will win the battle.

To introduce advanced CDMA systems which can improve capacity of SSMA. In particular, we will focus on cancellation of co-channel interference (CCI) in SSMA which restricts the number of available users who can simultaneously access a channel or the capacity of SSMA[1]. Before discussing CCI cancellation, we will briefly review optimum receiver for SSMA from information theoretical viewpoint. We will discuss optimum and sub-optimum multi-user detection for asynchronous SSMA systems, in which transmission of every user is not synchronized one another.

In wireless personal, indoor and mobile communication channels, it is difficult to carry out optimum detection for asynchronous SSMA due to unknown and time-varying characteristics of the channels. Therefore, sub-optimum multi-user detection based on CCI cancellation is more attractive. Since CCI of DS/SSMA and that of FH/SSMA have different origins, we will discuss several schemes of CCI cancellation based on temporal filtering, spatial filtering and coding techniques for DS/SSMA and FH/SSMA individually.

2. SSMA System Model

In this section, a model of SSMA is described in order to clarify a problem in SSMA. For the sake of simplicity, we formalize a DS/SSMA system model.

Fig. 1 shows a model of DS/SSMA system, where M users are transmitting individually spread spectrum signals $S_m(t), m = 1, 2, \cdots, M$ at the same time and in the same frequency band. This system can also be considered as the reverse link or uplink of a single cell in a cellular mobile communication system.

All the multiplexed SS signals except the desired SS signal interfere with the desired SS signal, due to the crosscorrelation of the different spreading sequences assigned to individual users, and is described as co-channel interference (CCI).

The output signal of the correlator for the nth user can be derived by integrating $r(t)C_n(t)$ over $T_d = NT$. When spreading sequence acquisition is performed, the sample value $y_{n,k}$ of the output at instant kT_d, can be represented in the form

$$\begin{aligned} y_{n,k} &= h_{n,0}d_{n,k} \qquad (1) \\ &+ \sum_{i\neq 0}\sum_{j=0}^{N-1} h_{n,j+iN}\Theta_{n,n}(j)d_{n,k-i} \\ &+ \sum_{i}\sum_{j=0}^{N-1}[\sum_{m=0}^{n-1} h_{m,j+iN}\{R_{n,m}(j+j_m)d_{m,k-i-1} + \\ &\quad \hat{R}_{n,m}(j+j_m)d_{m,k-i}\} \\ &+ \sum_{m=n+1}^{M} h_{m,j+iN}\{R_{n,m}(j+j_m)d_{m,k-i} + \\ &\quad \hat{R}_{n,m}(j+j_m)d_{m,k-i+1}\}] \\ &+ i_{n,k} \\ &+ n_{n,k} \end{aligned}$$

where $h_{m,j} = h_m(jT)$. It is assumed that $j_m = (t_{n,0} - t_{m,0})/T \bmod N$. $R_{n,m}(j)$ and $\hat{R}_{n,m}(j)$ are the partial crosscorrelation functions between the spreading sequences of the n-th and the m-th user [1].

In (6), the first term of the right hand side indicates the desired baseband data, the second term, the intersymbol interference (ISI) in the desired user's signal, the third, the co-channel interference (CCI) due to the undesired user, the fourth, the interfering signal component, and the fifth, the noise component. The magnitude of the co-channel interference is determined by the periodic or even crosscorrelation function $\Theta_{n,m}(j) = R_{n,m}(j) + \hat{R}_{n,m}(j)$ and $\hat{\Theta}_{n,m}(j) = R_{n,m}(j) - \hat{R}_{n,m}(j)$ depending on $d_{m,k} = d_{m,k-1}$ and $d_{m,k} = -d_{m,k-1}$, respectively.

3. Multi-User Detection for SSMA

Note from the above-discussion that CCI due to crosscorrelation of spreading sequences hampers establishment of acquisition and tracking, and limits the capacity of SSMA. CCI results in the "near-far " problem. Therefore, SSMA receivers should be designed so as to improve the synchronization and the capacity against the near-far problem.

3.1 Single-User Receivers and Multi-User Receivers

In general, SSMA receiver structures can be classified as either multi-user or single-user structures.

The conventional single-user detector assigned to each user is described in the previous section. Optimum single-user detector was proposed by Poor *et al.*[2] The conventional single-user receiver decides the correlator output for each user only in the interval corresponding to the data symbol individually among multiple-access users. In the decision, CCI is considered as an undesired component like noise.

From an information theoretical viewpoint, however, we can utilize CCI as redundant information which multiple-access users are sharing in a common channel or a multi-user channel, in order to afford performance gains over the conventional single-user receiver. The receiver by which all multiple-access users' signals can be correctly detected using CCI is named the multi-user receiver. The multi-user receiver can be used at a cell site or a base station to detect all multiple-access users' signals.

In contrast with the single-user receiver, for multi-user receivers, the decision of all the users' data is an inter-dependent process. Although their structures are much more complicated, their BER performance can greatly exceed that of the conventional correlator. These receivers vary in complexity from the optimum multi-user detector presented independently by Kohno *et al.* [3][4] and Verdu [5] to the suboptimum detectors developed independently by Kohno *et al.* [6][7], Masamura, Xie *et al.* [8], and Varanasi *et al.* [9].

Apart from [7], these detectors were all designed for the AWGN channel. On the other hand, the receiver of [7] which adaptively cancels the CCI was designed for time-varying baseband channels. Yoon *et al.* proposed multi-user receiver which can be considered as a continuation of the research of [7], and introduced a cascade of CCI cancellers and its application to multipath fading channels.

The above-mentioned receivers have been discussed mainly for usual DS/SSMA in which all simultaneous transmitted signals are received as the sum of their envelopes. It is named an *ADDER channel.* On the otherhand, FH/SSMA based on multilevel FSK developed by Goodman *et al.* [12] is considered a *OR channel,* whose receiver has no information erasure due to frequency collision or hit because an energy detector outputs total energy of multiple access users' transmitted signals in individual frequency bands. For such a *OR channel,* we can improve the capacity of SSMA by applying appropriate error-correcting coding

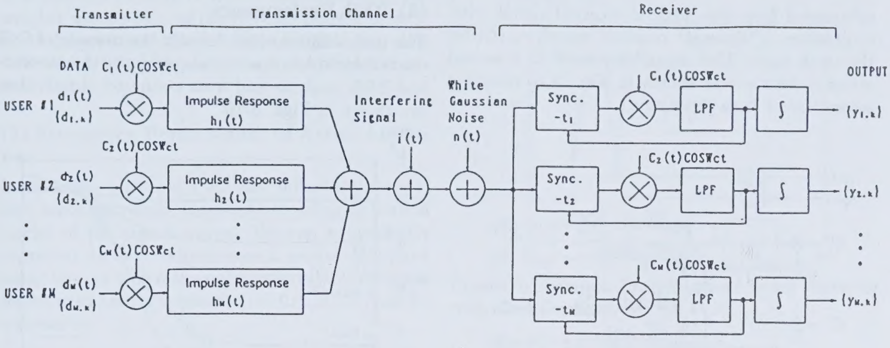


Figure 1: A Model of a DS/SSMA System

and decoding[13] and cancelling CCI different from them for *ADDER channel*. These techniques will be introduced in Section 5.

3.2 Optimal Multi-User Receiver

Without loss of generality and for the sake of notational simplicity, (6) can be rewritten if there is no intersymbol interference or multi-path distortion in a channel, no narrowband interference and only two users accessing the channel.

$$y_{1,k} = h_{2,0}R_{1,2}\mathbf{d}_{\mathbf{2,k-1}} + h_{1,0}\mathbf{d}_{\mathbf{1,k}} + h_{2,0}R_{2,1}\mathbf{d}_{\mathbf{2,k}} + n_{1,k} \quad (2)$$

$$y_{2,k} = h_{1,0}R_{2,1}\mathbf{d}_{\mathbf{1,k}} + h_{2,0}\mathbf{d}_{\mathbf{2,k}} + h_{1,0}R_{1,2}\mathbf{d}_{\mathbf{1,k+1}} + n_{2,k} \quad (3)$$

where $R_{1,2}$ and $R_{2,1}$ are partial crosscorrelation factors. It is noted from these equations that the correlator outputs $y_{1,k}$ and $y_{2,k}$ can be considered as multi-dimensional finite-state machine outputs corrupted with noise or multi-dimensional analogue convolutional encoder output with noise. It can be illustrated by its equivalent multi-dimensional tapped delay lines. Therefore, we can achieve optimum multi-user detection for SSMA by using a Viterbi algorithm that is well known as an efficient maximum likelihood sequence detection for a convolutional code, a trellis coded modulation and so on. In general, the optimum K-user receiver consists of a bank of single-user correlators in Fig. 1 followed by a Viterbi algorithm whose complexity per binary decision is $O(2^K)$ [3]-[5].

4. Co-Channel Interference Cancellation for DS/SSMA

The optimum multi-user receiver affords the much performance gain over conventional single-user one, but expends some possible disadvantages such as higher complexity and the need to know the relative characteristics of the multi-user channel, e. g. amplitudes, path delays and path phases in (5).

In this section, CCI cancellation techniques which carry out sub-optimum but more practical multi-user receivers will be explained. There are several methods which strive to reduce or remove CCI. The design of spreading sequences attempts to produce sets of sequences with optimum correlation properties [1][14]. To further aid the receiver in cancelling CCI, temporal, spatial signal processing and their combination can be employed.

4.1 CCI Cancellation by Temporal Adaptive Filtering

(1) Single Stage CCI Canceller

The receiver must obtain initial data estimates for use in CCI cancellation. We simply perform a hard decision upon the k-th user's correlator output $y_{k,n}$ in (6) as shown in Fig. 1; $\hat{b}_n^{(k)} = sgn(y_{k,n})$. Each initial data estimate, $\hat{b}_n^{(k)}$, of the k-th user's originally transmitted data $d_{k,n}$ at instant $nT_b(T_b = T)$, is first multiplied by its respective synchronized spreading sequence and carrier signal to create a replica of the originally transmitted signal as shown in Fig. 2. Then the signal is passed through a transversal filter(TF) which emulates channel $h_k(t)$. The estimation of the channel parameters may be feasible by an adaptive updating algorithm [7][10][11].

The output of the filter consists of two signals: the sum, $F^{(k)}$, of the kth user's faded terms ($\lambda \geq 2$) and the full replica of his received signal component which is the sum of $F^{(k)}$ and his first path's signal ($\lambda = 1$), $D^{(k)}$. By adding the m-th user's own faded terms, $F^{(m)}$, with all the other users' $F^{(k)} + D^{(k)}$, the canceller can create a replica of the CCI of the m-th user (including his intersymbol interference). The CCI replica of each user is subtracted from the original received signal, $r(t)$, to produce a "cleaned" received signal, $r_m(t)$, for the m-th user. This signal is passed to a second set of correlators as shown in Fig. 3 to obtain a second set of data estimates.

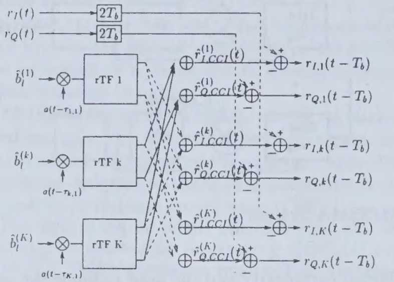

$r_I(t)$
$r_Q(t)$
$2T_b$
$\hat{b}_l^{(1)}$
rTF 1
$a(t-\tau_{1,1})$
$\hat{b}_l^{(k)}$
rTF k
$a(t-\tau_{k,1})$
$\hat{b}_l^{(K)}$
rTF K
$a(t-\tau_{K,1})$
$\hat{r}_{I,CCI}^{(1)}(t)$
$\hat{r}_{Q,CCI}^{(1)}(t)$
$\hat{r}_{I,CCI}^{(k)}(t)$
$\hat{r}_{Q,CCI}^{(k)}(t)$
$\hat{r}_{I,CCI}^{(K)}(t)$
$\hat{r}_{Q,CCI}^{(K)}(t)$
$r_{I,1}(t-T_b)$
$r_{Q,1}(t-T_b)$
$r_{I,k}(t-T_b)$
$r_{Q,k}(t-T_b)$
$r_{I,K}(t-T_b)$
$r_{Q,K}(t-T_b)$

Figure 2: CCI Canceller

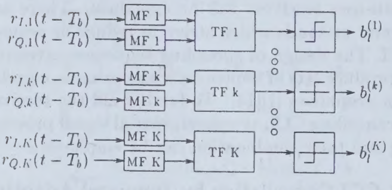

$r_{I,1}(t-T_b)$
$r_{Q,1}(t-T_b)$
MF 1
TF 1
$b_l^{(1)}$
$r_{I,k}(t-T_b)$
$r_{Q,k}(t-T_b)$
MF k
TF k
$b_l^{(k)}$
$r_{I,K}(t-T_b)$
$r_{Q,K}(t-T_b)$
MF K
TF K
$b_l^{(K)}$

Figure 3: Data Decision

(2) Cascade of CCI Canceller

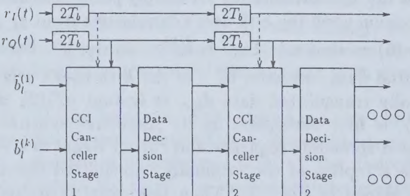

$r_I(t)$
$r_Q(t)$
$2T_b$
$\hat{b}_l^{(1)}$
$\hat{b}_l^{(k)}$
$\hat{b}_l^{(K)}$
CCI Canceller Stage 1
Data Decision Stage 1
CCI Canceller Stage 2
Data Decision Stage 2

Figure 4: Cascade of CCI Cancellers

Next, a cascade of such CCI canceller and data decision stages (Figs. 2 and 3) can be constructed as shown in Fig. 4. The decisions of the final stage give the final data decisions for the receiver. Intuitively, if the number of incorrect data decisions at the first stage in comparison to that at the initial data estimation has been reduced, more accurate replicas can be reconstructed at the CCI canceller of the second stage. In repeating this process, at a certain stage, the data decisions should converge to the actually transmitted data values.

(3) BER Performance

The probabilities of bit error in the cascade of CCI cancellers which was calculated from the theoretical BER analysis and from computer simulations are shown in Figs. 8-10.

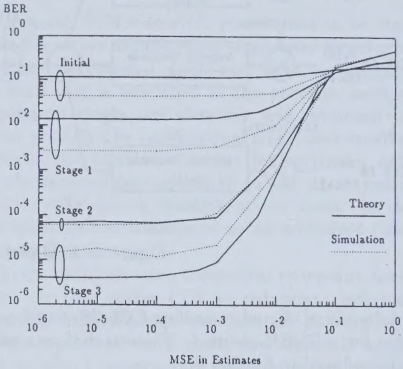


Figure 5: BER vs. MSE with $K = 64$ and $E_b/N_o = 10$ dB for up to three canceller stages. $\bar{L} = 2$ and $N = 127$.

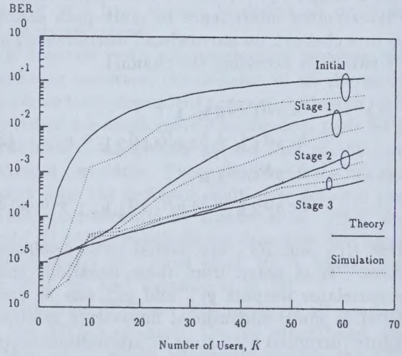


Figure 6: BER vs. K with $E_b/N_o = 10$ dB and MSE $= 10^{-2}$ for up to three canceller stages. $\bar{L} = 2$ and $N = 127$.

4.2 CCI Cancellation by Spatial Adaptive Filtering

An adaptive array antenna is useful in suppressing interfering signals because it can adaptively control directivity of the antenna by updating weights of element antennas even if the desired signal's arrival angle is unknown. It is usually assumed that the reference signal is correct and the array can suppress interference effectively enough to obtain correct reference signal. However, since the weights are updated by the error signal or the difference between the reference signal and the antenna output which includes residual interference and noise, it is difficult to achieve stable convergence of the weights when power of interference and/or noise is much larger than that of desired signal, i.e. low DIR and DNR, which are usual in a SSMA channel.

(1) Frequency Performance of Array Antennas

In an array antenna, there are different time delays among element antennas according to arrival angles of the signal waves. We can equivalently represent an array antenna as a model of tapped delay line, in which the delay intervals T_i of signal waves with arrival angles θ_i $(i = 0, 1, 2, ...I)$ can be written by

$$T_i = \frac{L}{c}\sin(\theta_i), \qquad (4)$$

where L is spatial interval among elements and c is a propagation speed of electric magnetic wave.

If arrival angles and received power of the signal waves are known, optimum weights of the array can be obtained by solving the Wiener-Hopf equation. Then the frequency performance of the array can be derived. Figs.12 and 13 show the frequency performance and the antenna pattern of array with two elements when there are two waves with arrival angles of 0 and 60 degree.

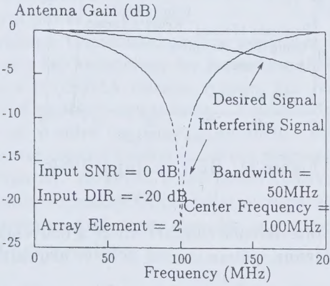


Figure 7: Temporal Frequency Performance of the Array Antenna with Two Elements

(2) TDL Array Antenna for Wideband Signals

A tapped delay line (TDL) array antenna was proposed to reject wide-band interference. Fig. 9 shows a structure of the TDL array antenna. Since TDL of each element antenna can more efficiently utilize correlation of received signals than a usual line array antenna, the frequency performance can be improved for wide-band signals such as interfering ss signals.

Fig. 10 shows the frequency performance of the TDL array antenna with two elements and five taps in an element. It is noted that Fig. 10 can reject wider band interference than Fig. 7.

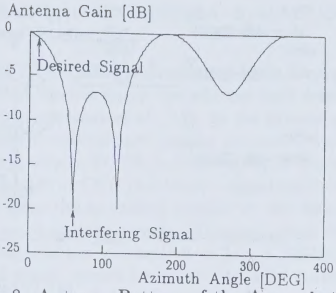


Figure 8: Antenna Pattern of the Array Antenna with Two Elements

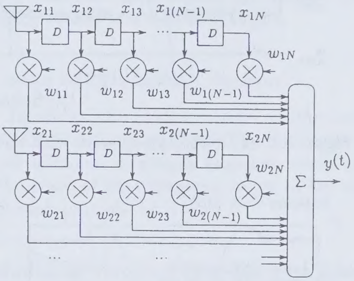


Figure 9: Structure of a Tapped Delay Line (TDL) Array Antenna

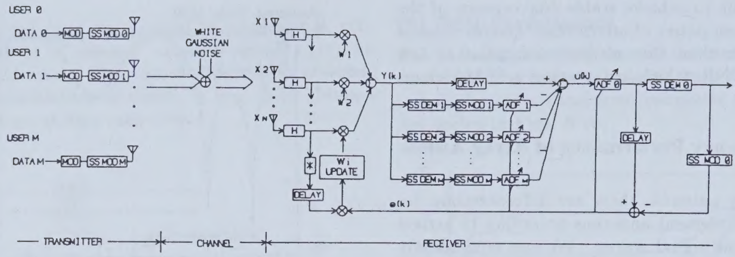
(3) System Evaluation

Computer simulations illustrate performance of the proposed TDL adaptive array antenna using the processing gain in comparison with conventional ones. In simulations, we assumed that there are desired and interfering SS signal waves with arrival angles of 20 and 60 degree, respectively, a carrier frequency of $100MHz$, spreading sequences with a chip rate of $25Mc/s$ and a period of $40nsec$, a data rate of $98Kb/s$, and a SS signal bandwidth of $50MHz$. The TDL array antenna has two elements and five taps in an element with a tap delay interval of $10nsec$ $(r = 2)$.

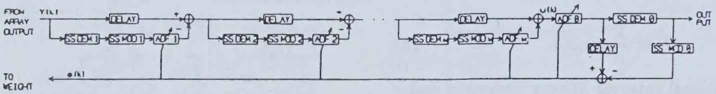
Figs.17 and 18 show BER performances vs. the input DNR and the input DIR, respectively. We can confirm that the proposed TDL adaptive array antenna using the processing gain improves the BER performance in low DNR and DIR.

4.3 Combination of Spatial and Temporal Filtering

If there is a high-level interfering SS signal from an undesired user with the same arrival angle as that of a desired user, an array antenna cannot suppress it. In order to solve this problem, combination of an adaptive array antenna in the spatial domain and an adaptive interference canceller in the tem-


USER 0
DATA 0
MOD
SS MOD 0
USER 1
DATA 1
SS MOD 1
USER M
DATA M
SS MOD M
WHITE GAUSSIAN NOISE
X 1
X 2
X N
H
W 1
W 2
Wi UPDATE
DELAY
e(k)
Y(k)
u(k)
SS DEM 1
SS MOD 1
ADF 1
SS DEM 2
SS MOD 2
ADF 2
SS DEM M
SS MOD M
ADF M
ADF 0
SS DEM 0
SS MOD 0
OUT PUT
TRANSMITTER
CHANNEL
RECEIVER

(a) Parallel Canceller structure


FROM ARRAY OUTPUT
Y(k)
DELAY
SS DEM 1
SS MOD 1
ADF 1
SS DEM 2
SS MOD 2
ADF 2
SS DEM M
SS MOD M
ADF M
u(k)
ADF 0
SS DEM 0
SS MOD 0
OUT PUT
TO WEIGHT
e(k)

(b) Serial Canceller Structure

Figure 13: A DS/SSMA System with an adaptive array antenna including a canceller of CCI

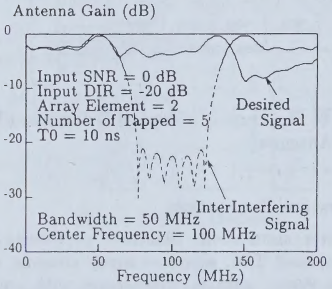


Figure 10: Temporal Frequency Performance of the TDL Array Antenna with Two Elements

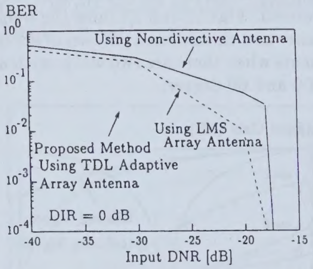


Figure 11: BER vs. Input DNR

poral domain was proposed [15]. The proposed system can suppress interfering SS signals,i.e. co-channel interference, with arrival angles different from that of a desired user by using a null steering array antenna and eliminate by means of a canceller the residual interference and co-channel interference having an arrival angle the same as that of the desired SS signal.

Fig. 13 shows the structure of the DS/SSMA system based on combination of adaptive spatial and temporal filtering, which consists of array antenna and canceller using adaptive digital filters (ADFs). Fig. 14 shows BER of demodulated data as a function of the D/I ratio for the same arrival angle. The proposed system can achieve stable demodulation and improve BER performance even in a heavy interference channel where a conventional array antenna system cannot achieve acquisition.

4.4 Combination of CCI Cancellation and Power Control

In the serial structure of the canceller shown in Fig. 13(b), interfering SS signals or CCIs from multiple access users are cancelled in the order of decreasing receiving power because it is easy to achieve acquisition, demodulation and cancellation of an interfering SS signal having greater power and its cancellation makes it possible or easy to cancel other interfering SS signals [15][16].

This discussion suggests that the performance of CCI cancellation or the capacity of SSMA depends upon receiving signal power distribution among multiple access users. In fact, when the receiving power distribution is exponential, the capacity

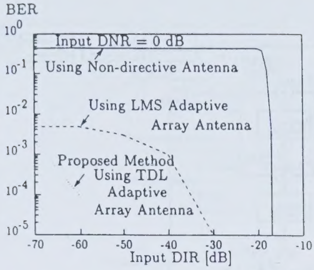


Figure 12: BER vs. Input DIR

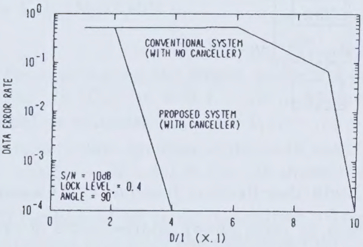


Figure 14: Demodulated BER vs. D/I Ratio

can be improved to be 1.5 times as large as that in uniform power distribution.

Although transmission power control is considered as a key technology for increasing the capacity in a DS/SSMA cellular system, the receiving power of multiple access users is controlled so uniform as to solve the near-far problem. Errors in the power control results in severe degradation of the capacity. However, if the power control is applied so as to make CCI cancellation more efficient, the power control errors can also be compensated by CCI cancellation.

5. Co-Channel Interference Cancellation for FH/SSMA

There is no near-far problem in an FH/SSMA system and its frequency diversity effect is useful to combat with fading in personal, indoor and mobile radio communications. However, coincidence or collision of hopping frequencies of desired and undesired SS signals causes co-channel interference, namely "hit ", which restricts the capacity of FH/SSMA.

Conventionally, the following several countermeasures against CCI in FH/SSMA have been studied.

5.1 Countermeasures for CCI

(1) Coding Schemes
(2) Multi-Hopping Schemes
(3) Hybrid DS/FH Schemes
(4) Spatial Filtering Schemes

5.1 Multilevel FSK/SSMA System Using Address Codes

The MFSK(Multilevel Frequency Shift Keying) / FH-SSMA system using the address code was proposed by Goodman et al. [12]. In the system using the address code as the hopping sequence, the carrier frequency in MFSK is hopped[19]. Even when there is a hit or CCI, the desired signal can be separated from the undesired signals by the decoding based on the address code. However, when many users' signals are overlapped or hit in SSMA, the desired signal cannot be always decoded correctly. As mentioned in **3.1**, since this multi-user channel is considered as a *OR channel*, we can improve the capacity of SSMA by applying appropriate error-correcting coding and decoding[13][20].

(1) Coded MFSK/SSMA System

Fig. 15 shows the system model of the coded MFSK/FH-SSMA system, and Fig. 18 shows an example of time - frequency matrices which indicate the state of the frequency tones of the each chip in each part of Fig. 15.

(2) Modified Decoding Scheme for a Multi-User Receiver in the Coded MFSK/SSMA

In conventional decoding, if there are several rows having the largest number of entries in one time - frequency matrix decoded by the address code, then the received symbol which corresponds to that matrix must be considered as an erasure and decoded by the conventional error and erasure correction decoding. The erasure is considered to be one among 2^K kinds of symbols with equal probability. In our modified decoding scheme, however, we use the information that indicates which symbols correspond to rows having the largest number of entries in the time-frequency matrix. Since we can reduce the number of possible symbols or candidates by the information, the error and erasure correcting capability can be improved and the user capacity can consequently increase[20].

Fig. 16 shows the theoretically calculated user capacity normalized by the code rate versus the code rate for the modified decoding scheme in comparison with the usual decoding scheme without erasure, the conventional error and erasure correction decoding scheme, and the maximum likelihood decoding scheme.

In the maximum likelihood decoding scheme, the user capacity is theoretically calculated from the channel cutoff rate without restricting the utilized error-correcting code. Therefore, the user capacity of the maximum likelihood decoding scheme using RS code is smaller than a bound of the user capacity derived from the channel cutoff rate because of the restriction of the length of RS code.

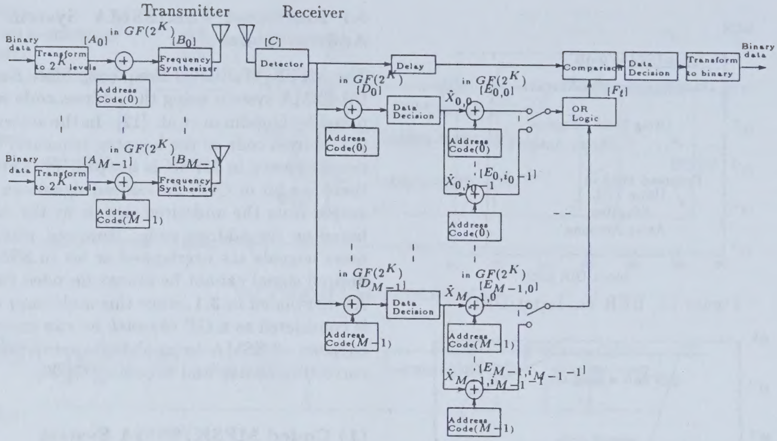

Transmitter
Receiver
Binary data
Transform to 2^K levels
$[A_0]$
in $GF(2^K)$
$[B_0]$
Frequency Synthesizer
Address Code(0)
$[A_{M-1}]$
$[B_{M-1}]$
Address Code($M-1$)
$[C]$
Detector
Delay
in $GF(2^K)$ $[D_0]$
Data Decision
Address Code(0)
in $GF(2^K)$ $[E_{0,0}]$
$\hat{X}_{0,0}$
$[E_{0,i_0-1}]$
$\hat{X}_{0,i_0-1}$
in $GF(2^K)$ $[D_{M-1}]$
Address Code($M-1$)
in $GF(2^K)$ $[E_{M-1,0}]$
$\hat{X}_{M-1,0}$
$[E_{M-1,i_{M-1}-1}]$
$\hat{X}_{M-1,i_{M-1}-1}$
Comparison
$[F_t]$
OR Logic
Data Decision
Transform to binary
Binary data

Figure 15: MFSK/FH-SSMA System Using the Proposed Multi-User Receiver Based on CCI Canceller

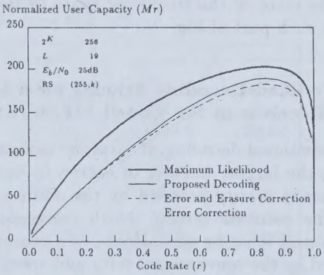


Figure 16: Normalized User Capacity versus Code Rate in the Coded MFSK/FH-SSMA System in a Rayleigh Fading Channel

5.3 Muti-User Receiver Based on CCI Cancellation for a MFSK/SSMA System

As an optimum multi-user receiver for the MFSK/SSMA, we have proposed and investigated a multi-user receiver in which all users' data are correctly decided by cancelling CCI, in order to improve user capacity of SSMA[21]. In decoding, we use the time-frequency matrices in which the desired and undesired signal tones are marked with *OR* logic. When in the decoding matrix there are not only a correct row corresponding to the desired user's data tone but also aliasing rows which consist of undesired users' signal tones and have no less entries than the correct row, the conventional decoding cannot correctly detect the desired user's data tone. However, since the proposed receiver can regenerate replicas of the undesired users' signal tones by using known address codes of every user, it can eliminate CCI by subtracting entries of the replicas from the decoding time-frequency matrix.

(1) System Configuration

Fig. 15 shows the model of MFSK/FH-SSMA system with the proposed multi-user receiver, and Fig. 18 illustrates the decoding and cancelling CCI using the time - frequency matrices which indicate the state of the frequency tones of the each chip in each part of Fig. 15.

(2) Muti-User Detection Based on CCI Cancellation

The process of decoding and cancelling CCI in MFSK/FH-SSMA system is described corresponding to Figs.21 and 22 as follows.
(A) The matrices $[D_m]$ $(m = 0, 1, \cdots, M-1)$ of Fig. 18 show each user's time - frequency matrices decoded by each user's address code $\mathbf{R}_m$ $(m = 0, 1, \cdots, M-1)$. $Z_{m,n,l}$ $(m = 0, 1, \cdots, M-1)$ shows an entry of the vectors decoded by the address code or an entry of $[D_m]$ $(m = 0, 1, \cdots, M-1)$. The subscript n denotes entries of n-th user over the time - frequency matrix. When $n = m$, that shows entries of the desired user.
(B) Let i_m $(1 \leq i_m \leq 2^K)$ be the number of rows having the largest number of entries in the time - frequency matrix decoded by the m-th user's address code. (e.g. there are two lows having five entries at $[D_0]$ in Fig. 18, i.e. $i_0 = 2$, and $i_m = 1$ $(m = 1, \cdots, M-1)$.) The symbols corresponding

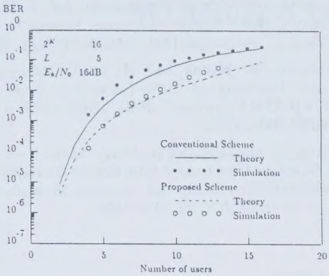


Figure 17: BER versus Number of users ($2^K = 16$, $L = 5$, E_b/N_0=20dB)

to the rows having the largest number of entries in Fig. 18 $[D_m]$ $(m = 0, 1, \cdots, M-1)$, are considered as estimates $\hat{X}_{m,j}$ $(j = 0, 1, \cdots, i_m - 1)$ of the correct data symbols of the m-th user. $[D_m^{'}]$ $(m = 0, 1, \cdots, M-1))$ in Fig. 18 shows the estimates.

(C) Replicas of all users' transmitted vectors are regenerated by adding the estimates with the address codes, for every combination of estimates of all the user's data symbols. The replica $W_{m,j,l}$ regenerated by the estimate $\hat{X}_{m,j}$ of the m-th user's data can be represented by $W_{m,j,l} = \hat{X}_{m,j} + R_{m,l}$ for $(m = 0, 1, \cdots, M-1), (l = 0, 1, \cdots, L-1), (j = 0, 1, \cdots, i_m - 1)$. $[E_{m,j}]$ in Fig. 18 shows the replicas regenerated by the estimate $\hat{X}_{m,j}$. These replicas $\hat{W}_{m,j,l}$ $(m = 0, 1, \cdots, M-1)$ are added in logical *OR* operation. Then, we can derived $\prod_{m=0}^{M-1} i_m$ candidates of the received matrices $[F_t](t = 0, 1, \cdots, \prod_{m=0}^{M-1} i_m - 1)$ in Fig. 18.

(D) If we select the $[F_t](t = 0, 1, \cdots, \prod_{m=0}^{M-1} i_m - 1))$ which has the most number of coincident entries with entries of the received matrix $[C]$, we can decode all users' data correctly.

(3) System Evaluation

Fig. 17 shows BER(Bit Error Rate) versus the number of user which is derived by the theoretical analysis and computer simulation for the MFSK/FH-SSMA system without and with cancellation of CCI.

Of course, if we jointly use the multi-user receiver based on CCI Cancellation and the modified Erasure Decoding mentioned in **5.2** for Coded MFSK/SSMA systems, optimum performance will be performed.

7. CONCLUDING REMARKS

For DS/SSMA and FH/SSMA, optimum and suboptimum multi-user receivers based on CCI cancellation have been discussed.

For wider commercial applications of SS techniques, i.e. consumer and personal communications, improving transmission capacity of SSMA and combating with narrowband interference and fading are required compatibly. For this purpose, the advanced DS and FH CDMA systems based on CCI cancellation must be a reasonable solution.

References

[1] R. Kohno, "Pseudo-Noise Sequences and Interference Cancellation Techniques for Spread Spectrum Systems," *IEICE Transactions on Communications, Electronics, Information and Systems,* vol.E74, No.5, May 1991.

[2] H. V. Poor and S. Verdu, "Single-User Detectors for Multiuser Channels," *IEEE Trans. Commun.*, vol. COM-36, pp. 50-60, Jan. 1988.

[3] R. Kohno, H. Imai and M. Hatori, "Cancellation Techniques of Co-Channel Interference in Asynchronous Spread Spectrum Multiple Access," *IECE Tech. Rep.*, CS82-38, pp.29-35, June 1982.

[4] R. Kohno, H. Imai and M. Hatori, "Soft Decision Receiver Using Viterbi Algorithm for Asynchronous SSMA," *Proc. IECE National Conference,* 1322, p.5-193, March 1983.

[5] S. Verdu, "Minimum Probability of Error for Asynchronous Gaussian Multiple Access Channels," *IEEE Trans. Inform. Theory,* vol. IT-32, pp. 85-96, Jan. 1986.

[6] R. Kohno, H. Imai and M. Hatori, "Cancellation Techniques of Co-Channel Interference in Asynchronous Spread Spectrum Multiple Access Systems," *Trans. IECE, Japan,* vol. J66-A no. 5, pp. 416-423, May 1983 (in Japanese).
Electronics & Communications in Japan, Scripta Publishing Company, vol. 66, no. 5, pp. 20-29, May 1983, (English version).

[7] R. Kohno, H. Imai, M. Hatori and S. Pasupathy, "An Adaptive Canceller of Co-channel Interference for Spread Spectrum Multiple Access Communication Networks in a Power Line," *IEEE J. Select. Areas Commun.*, vol. SAC-8, pp. 691-699, May 1990.

[8] Z. Xie, R.T. Short, C.T. Rushforth, "A Family of Suboptimum Detectors for Coherent Multiuser Communications," *IEEE J. Select. Areas Commun.*, vol. SAC-8, pp. 683-690, May 1990.

[9] M. K. Varanasi, B. Aazhang, "Multistage Detection in Asynchronous Code-Division Multiple-Access Communications," *IEEE Trans. Commun.*, vol. COM-38, pp. 509-519, April 1990.

[10] Y. C. Yoon, R. Kohno and H. Imai, "Cascaded Co-Channel Interference Cancelling and Diversity Combining for Spread-Spectrum Multi-Access over Multipath Fading Channels," *IEICE Trans. Commun., Japan,* vol. E76-B no. 2, Feb. 1993.

[11] Y. C. Yoon, R. Kohno and H. Imai, "A Spread-Spectrum Multi-Access System with Co-Channel Interference Cancellation over Multipath Fading Channels," *IEEE J. Select. Areas Commun.*, (to be published) Dec. 1993.

[12] D. J. Goodman, P. S. Henry and V. K. Prabhu, "Frequency-hopped Multilevel FSK for Mobile Radio," *Bell Syst. Tech. J.*, vol.59, No.7, pp.1257-1275, Sept. 1980.

[13] T. Mabuchi, R. Kohno and H. Imai, "An Advanced Decoding Scheme of a Coded MFSK/FH-SSMA System," *IEICE Tech., Rep.*, SST92-25, June 1992.

[14] D. V. Sarwate and M.B.Pursley, "Crosscorrelation Properties of Pseudorandom and Related Sequences," *Proc. IEEE,* vol.68, pp.593-619, 1980.

[15] R. Kohno, H. Imai, M. Hatori and S. Pasupathy, "Combination of an adaptive Array Antenna and a Canceller of Interference for Direct-Sequence Spread-Spectrum Multiple-Access System," *IEEE J. Selected Areas Comun.*, JSAC8, 4, pp.675-682, May 1990.

[16] A. J. Viterbi, "Very Low Rate Convolutional Codes for Maximum Theoretical Performance of Spread-Spectrum Multiple-Access Channels," *IEEE J. Select. Areas Commun.*, vol. JSAC-8, pp. 641-649, May 1990.

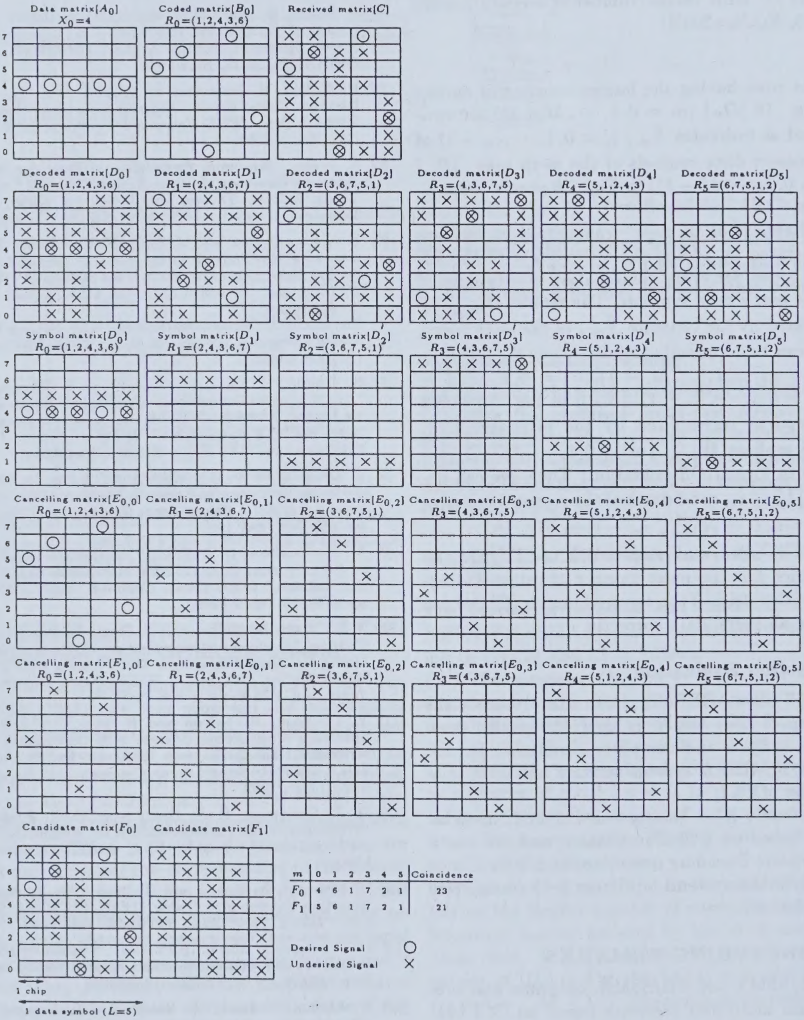
[17] H.Yamamura, R.Kohno and H.Imai, "Interference Cancellation and Capacity of a Cellular CDMA System Based on Multihop Fast Frequency Hopping," *Proc. IEEE Vehicular Technology Conference,* pp.71-76, May 1991

[18] T. Mabuchi, R. Kohno and H. Imai, "A Note on Cancelling Cochannel Interference in Frequency Hopping Multiple Access (FH/SSMA)," *IEICE Tech. Rept.*, SSTA91-12, pp.75–82, March 1991.

[19] G. Einarsson, "Address Assignment for a Time-Frequency-Coded, Spread-Spectrum System," *Bell Syst. Tech. J.*, vol.59, pp. 1241-1255, No.7, Sept. 1980.

[20] T. Mabuchi, R. Kohno and H. Imai, "Multihopping and Decoding of Error-Correcting Code for MFSK/FH-SSMA Systems", *Proc. IEEE ISSSTA'92*, pp.199-202, Nov. 1992.

[21] R. Kohno, T. Mabuchi and H. Imai, "Multi-User Receiver Based on Co-Channel Interference Cancellation for Multilevel FSK/FH-Spread Spectrum Multiple Access," *Proc. IEEE ICC'93*, May 1993.


Data matrix[A_0] $X_0=4$
Coded matrix[B_0] $R_0=(1,2,4,3,6)$
Received matrix[C]
Decoded matrix[D_0] $R_0=(1,2,4,3,6)$
Decoded matrix[D_1] $R_1=(2,4,3,6,7)$
Decoded matrix[D_2] $R_2=(3,6,7,5,1)$
Decoded matrix[D_3] $R_3=(4,3,6,7,5)$
Decoded matrix[D_4] $R_4=(5,1,2,4,3)$
Decoded matrix[D_5] $R_5=(6,7,5,1,2)$
Symbol matrix[D_0'] $R_0=(1,2,4,3,6)$
Symbol matrix[D_1'] $R_1=(2,4,3,6,7)$
Symbol matrix[D_2'] $R_2=(3,6,7,5,1)$
Symbol matrix[D_3'] $R_3=(4,3,6,7,5)$
Symbol matrix[D_4'] $R_4=(5,1,2,4,3)$
Symbol matrix[D_5'] $R_5=(6,7,5,1,2)$
Cancelling matrix[$E_{0,0}$] $R_0=(1,2,4,3,6)$
Cancelling matrix[$E_{0,1}$] $R_1=(2,4,3,6,7)$
Cancelling matrix[$E_{0,2}$] $R_2=(3,6,7,5,1)$
Cancelling matrix[$E_{0,3}$] $R_3=(4,3,6,7,5)$
Cancelling matrix[$E_{0,4}$] $R_4=(5,1,2,4,3)$
Cancelling matrix[$E_{0,5}$] $R_5=(6,7,5,1,2)$
Cancelling matrix[$E_{1,0}$] $R_0=(1,2,4,3,6)$
Cancelling matrix[$E_{0,1}$] $R_1=(2,4,3,6,7)$
Cancelling matrix[$E_{0,2}$] $R_2=(3,6,7,5,1)$
Cancelling matrix[$E_{0,3}$] $R_3=(4,3,6,7,5)$
Cancelling matrix[$E_{0,4}$] $R_4=(5,1,2,4,3)$
Cancelling matrix[$E_{0,5}$] $R_5=(6,7,5,1,2)$
Candidate matrix[F_0]
Candidate matrix[F_1]
7
6
5
4
3
2
1
0
1 chip
1 data symbol (L=5)
Desired Signal ○
Undesired Signal ×

m	0	1	2	3	4	5	Coincidence
F_0	4	6	1	7	2	1	23
F_1	5	6	1	7	2	1	21

Figure 18: An Example of Decoding with CCI Cancellation ($2^K = 8$, $L = 5$, $M = 6$; five users CDMA)

WIRELESS DATA COMMUNICATIONS

Kaveh Pahlavan

Center for Wireless Information Studies
Electrical and Computer Engineering Department
Worcester Polytechnic Institute
Worcester, MA 01609

Tel/fax: (508) 831-5634/5491

Abstract

This tutorial addresses issues related to wireless data communications. Overview of the emerging wireless information networks for voice and data applications. Wireless LANs and MANs: market, history and standards. Wireless LANs: topologies and technologies - a comparative study. Research topics: wireless-specific experimental applications, wireless access in an integrated common band, building-specific radio propagation modeling, CAD facilities for analysis, design and installation of wireless networks, ASIC designs for wireless applications.

Introduction

The new-generation wireless information networks are evolving around either voice-driven applications such as digital cellular, cordless telephone, and wireless PBX; or around data-driven networks such as wireless LANs and mobile data networks. While it is true that all the major standards initiatives are focused on integration of services, one still sees a separation of the industrial communities that participate in the various standards bodies. That is, we see GSM, IS-54, TR45.3, DECT and others supported primarily by representatives of the voice-communications industry, while IEEE 802.11, WINForum and HIPERLAN are supported primarily by those with interest in data communications. Figure 1 shows various dimensions of the voice and data communication industries and compares the local cordless personal communication with the wide area cellular and the WLLANs with the wide area low speed data services.

Although future personal devices may be designed as integrated units for personal computing as well as personal voice and data communications, the wireless access supporting different applications may use different frequency bands or even different technologies. A personal communications service may use a shared wideband-CDMA band, and digital cellular service another TDMA or CDMA band; while low-speed mobile data is carried in the gaps between bursts of voice activity, and high speed local-area data uses another shared wideband channel. At the same time, the various services may all be integrated in a metropolitan-area or wide-area network structured with ATM switches. The future direction of this industry depends upon technological developments, and a maturity in the spectrum-administration organizations, who must understand the growing massive demand for bandwidth and must in turn develop strategies allowing a fair sharing of increasingly scarce bandwidth.

Wireless Data: Market and User Perspectives

From the data user's perspective, the minimum satisfactory requirement for personal communication is low-speed access in wide areas and high-speed access in local areas. The low-speed wide-area access will serve a variety of short-message applications such as notice of electronic or voice mail, while the local-area access will support high-speed local applications such as long file transfers or printing tasks. In the current literature, low-speed wide-area wireless data communication is referred to as mobile data, while local high-speed data communication systems are called wireless LANs.

Mobile data

Mobile data radio systems have grown out of the success of the paging-service industry and increasing customer demand for more advanced services. Today 100,000 customers are using mobile data services and the industry expects 13 million users by the year 2000. This could account for 10 to 30 percent of the revenue of the cellular radio industry. Today, mobile data services provide length-limited wireless connections with in-building penetration to portable users in metropolitan areas. The future direction is toward wider coverage, higher speeds and capability for transmitting longer data files.

The data rates of existing mobile radio systems are comparable to voice-band modem rates (less than 19.2kbps). However, the service has a limitation on the size of the file that can be transmitted in each communication session. The coverage of the service is similar to the mobile radio services with the difference that the mobile data service must have in-building penetration. Mobile radio users typically use the telephone unit inside the vehicle and usually while driving. Mobile data users typically use the portable unit in a building and in a fixed location. Therefore in-building penetration is an essential feature of mobile data services.

Wireless LANs

Today, most large offices are equipped with wiring for LANs, and the inclusion of LAN wiring in the planning of a new office building is increasingly being done as a standard procedure, along with planning for telephone and electric-power wiring. The WLLAN market will very likely develop on the basis of the appropriateness of the wireless solution to specific applications. The targeted markets for the wireless LAN industry include applications on manufacturing floors, in offices with wiring difficulties, and in branch offices and temporary offices. The WLLAN industry expects to capture 5-15% of the LAN market in the near future.

Although the market for PCs is not growing as it has in past years, the market for portable computers such as lap-tops, pen-pads and notebook computers is growing rapidly. Of greater importance to the wireless data communication industry, the market for networked portable is growing much faster than the market for portable computing. Obviously, wireless is the communication method of choice for portable terminals. Mobile data communication services discussed earlier provide a low speed solution for wide area coverage. For high speed and local communications, a portable terminal with wireless access can bring the processing and data base capabilities of a large computer directly to specific locations for short periods of time, thus opening a horizon for new applications.

Frequency Administration Issues

The principal technological problems for implementation of wireless LANs are: (1) data rate limitations caused by the multipath characteristics of radio propagation, (2) the difficulties associated with signal coverage within buildings, and (3) the need for low-power electronic implementations suitable for portable terminals. These technical difficulties can be resolved and effective solutions are being developed for all of them. The greatest obstacle for the wireless multi-mega bit data communication services is the lack of a suitable frequency band for reliable high-speed communication. The existing ISM bands assigned for multiple-user applications are suitable for wireless LANs; however, they suffer from careless users who cause unnecessary interference to other users, and they are restricted to spread-spectrum technology. More widespread use of high speed wireless data communication technology will depend upon cooperation from frequency administration organizations in providing wider bandwidth allocations without restriction on the adopted technology, and in administering rules and etiquette for ethical use of these bands.

At frequencies around several GHz the technology is available for implementation with a reasonable size, power consumption, and cost. Moving to higher frequencies is the solution for the future. As the frequency increases, the prospect for obtaining a wider bandwidth from spectrum regulation agencies will improve. However, with today's technology, implementation at a few tens of GHz with reasonable product size and power consumption is challenging, particularly when wideband portable communication is considered. At higher frequencies signal transmission through walls is more difficult. For frequencies around a few tens of GHz, the signal is mostly confined within the room. This feature is advantageous in certain applications, such as military applications, where confinement of the signal within a room or building is a desirable security feature. Also, at higher frequencies the relationship between cell boundaries and the physical layout of the building is more easily determined, facilitating the planning of cell assignments within the building. The technology at higher frequencies is highly specialized and not commonly available within the computer industry. This has encouraged joint ventures among semiconductor, radio and computer companies to develop new products at these frequencies.

Existing WLLANs

The existing technologies for wireless LANs are the licensed cellular systems operating at 18-19GHz, unlicensed spread spectrum systems operating in ISM bands, and diffused and directed-beam infrared systems. Table I summarizes the features of the systems currently available in the market. Diffused IR provide moderate data rate and coverage. It is suitable for moderate-size offices, short-distance battery oriented applications (portable to printer), and environments with significant amounts of radio interference. Directed-beam IR offers higher speed with reasonable coverage for applications with fixed terminals. It is suitable for large-file transfers between main frames and servers as well as large open offices with many fixed terminals. WLLANs at tens of GHz are suitable for high speed communications in large partitioned areas such as larger offices or some libraries. The spread spectrum systems provide the largest coverage and are suitable for penetration through building floors. Spread spectrum technology is suitable for small-business applications where a few terminals are distributed over several floors of a building. An important factor in selection is the ease of installation and flexibility of the software to accommodate various protocols for interconnecting with the backbone wired LANs. The evolving next generation of wireless LANs is designed to be incorporated into the lap-top, note-book and pen-pad computers, requiring significant reduction of the size and power consumption for existing LAN products. It will be ideal if these devices also provide low speed wireless access for wide area coverage.

Spread Spectrum for WLLANs

Ever since that FCC announced ISM bands in May of 1985, various spread spectrum commercial products, from low speed fire safety devices to high speed wireless LANs, poured in the market. Today, both voice driven and data driven wireless information network industries are involved in spread spectrum applications. But, their point of view toward spread spectrum is quite different.

The voice-driven digital cellular and personal communication industries are considering code division multiple access (CDMA) spread spectrum as an alternative to time division multiple access (TDMA). All the studies in this industry is directed toward the bandwidth efficiency and design complexity of the CDMA as it is compared with the TDMA. Since the interface to the telephone network at different locations is an important aspect of the wireless voice driven services, manufacturers of digital cellular and personal communication devices must wait for the finalization of standards before they market their products. In contrast, to develop and market a wireless LAN, manufacturers only need a suitable band for high speed data communications, and the existence of a standard is not necessary. WLLANs are using spread spectrum because the first bands available for high speed data communication were ISM bands which are specifically assigned to spread spectrum technology.

At 902-928 MHz ISM band, 26 MHz bandwidth is available with a minimum bandwidth expansion rate of 11. The designers of WLLAN products at this band usually sacrifice bandwidth expansion to achieve higher data rates. This way one can achieve a data rate of around 2 Mbps with a commonly used QPSK modem. The ISM bands at higher frequencies offer wider bands and consequently higher data rates. Products operating at data rate above 5 Mbps use the 2.4 and 5.7 GHz ISM bands. At these frequencies, the coverage of the signal is more restricted.

An important feature of the spread spectrum technology for the WLLAN industry is the anti-multipath nature of the technique that increases the coverage and reliability of the modem. health-related issues in high-power transmission at microwave frequencies, The spread spectrum devices at 910MHz bands can cover several floors of a building, a feature which is not matched by other technologies.

Today, an important issue for the WLLAN industry is to provide services for the portable devices that requires low electronic power consumption for battery operation. In ISM bands, slow frequency hopping is considered for this purpose.

Evolving Mobile Data Networks

The mobile data networks are either designed as a private network only for data communications or they are integrating the data with a mobile radio network. The existing private mobile data networks are ARDIS and Mobitex evolved for data communication applications. The emerging networks using mobile radio infrastructure are CDPD and the IS-95 data channels. CDPD intends to use the existing cellular mobile bands and infrastructure (building and antenna sites) to transmit data in the free channels. The CDPD is a packet radio network using the free AMPS channels for data communications to provide a non-interfering service with the existing analog mobile radio system. The IS-95 working on standards for digital cellular using CDMA is considering data channels with data rates up to 9600 bps spreaded over the 1.23 Mcps. ETSI in Europe is also working on similar packet data services operating over the evolving digital cellular and personal communication services. Table II provides a comparison among the above services.

Research in Wireless Data Communications

Following is the list of some topics related to wireless networks demanding further research:

- Development of a fair "etiquette" for wireless access in an integrated voice/data network sharing a common band.
- Design of high speed modems.
- Channel simulation using building-specific radio propagation models.
- CAD facilities for design and analysis of wireless networks.
- Computation using wireless media.

Research in radio propagation modeling is needed for two purposes, (1) to evaluate various alternatives for the design of the system, (2) to determine the optimum location for the installation of the antennae in a large area. Research in this area started with measurement base statistical analysis of the received power [2,3] and the RMS multipath spread of the channel [1]. It followed by development of measurement based statistical models for radio propagations [5-10]. These measurements were used for performance evaluation of various modems working over the indoor radio channel [12-15]. More recently development of software for building specific radio propagation studies are under investigations [4,11]. Building specific radio propagation models are very helpful in installation of the antenna sites. Various methods of regenerating the time response of the indoor radio measurements are described in [9-14]. A method of regenerating the frequency response of indoor radio propagation is reported in [15-17].

If wireless LANs are to influence the LAN market, their data rate limitations has to be improved. The maximum data rate of simple modulation techniques in an indoor channel is around 1 Mbps [13]. The estimated data rate of 1 Mbps is rather low for wireless LANs. The above estimates improve for an order of magnitude by providing diversity channels for the receiver, employing modulation and coding techniques with greater bandwidth efficiency, using multi-rate modems, or implementing adaptive equalization techniques [11-15]. A comparative study among different alternatives has been an active area of research. With the recent FCC announcement on releasing new bands for data communications, further studies in these areas will continue. The optical wireless communications are also candidates for high speed data communications [16-18].

Analysis of the effects of fading on the performance of the throughput of the networks [19-20]. Analysis of the integration of voice and data networks [21] is also an interesting research topic.

References

1. Devasirvatham D.M.J., Time delay spread measurements of wideband radio systems within a building, Electronics Letters 20, 8 (1984), 949-950.
2. S.E. Alexander, " Characterizing buildings for propagation at 900 MHz," IEE Electronics Letters, Sept. 1983, pp 860.
3. S.E. Alexander S.E., "Radio propagation within buildings at 900 MHz", IEE Electronics Letters, Oct. 1982, pp 913-914.
4. T. Holt, K. Pahlavan and J.F. Lee, "A Graphical Indoor Radio Channel Simulator Using 2-D Ray Tracing", IEEE PIMRC, Boston, MA, Oct. 1992.
5. A.M. Saleh and R. Valenzuela, "Statistical Model for Indoor Multipath Propagation," IEEE JSAC, Feb. 1987, pp. 128-137.
6. R. Ganesh and K. Pahlavan, "On the Arrival of the Paths in Multipath Fading Indoor Radio Channels", IEE, Electronic Letters, June 1989,pp763-765.
7. T.S. Rappaport, S.Y. Seidel and K. Takamizawa, " Statistical Channel Impulse Response Models for Factory and Open Plan Building Radio Communication System Design," IEEE Trans. on Comm., May 1991, pp.798-806.
8. R. Ganesh and K. Pahlavan, "Modeling of the Indoor Radio Channel", IEE Proceedings-I, June 1991.
9. P. Yegani and C.D. McGillem, "A Statistical Model for the Factory Radio Channel", IEEE Trans. on Comm. Oct. 1991, pp. 1445-1454.

10. S. J. Howard and K. Pahlavan, "Autoregressive Modeling of Wideband Indoor Radio Propagation", IEEE Trans. on Comm., Sep. 1992.
11. T.A. Freeburg, "Enabling Technologies for Wireless In-Building Network Communications - Four Technical Challenges, Four Solutions," IEE Comm. Mag.,April 1991, pp 58-64.
12. G. Yang and K. Pahlavan, "Comparative Performance Evaluation of Sector Antenna and DFE Systems in Indoor Radio Channels", Proceedings of the IEEE ICC, Chicago, IL, June 1992.
13. T. A. Sexton and K. Pahlavan, "Channel Modeling and Adaptive Equalization of Indoor Radio Channels", IEEE Jour. of Sel. Areas in Comm. (JSAC), Jan. 1989.
14. K. Pahlavan, S. Howard, and T. Sexton,"Adaptive Equalization of Indoor Radio Channel", IEEE Trans. on Comm., Jan. 1993.
15. A.S. Acampora and J.H. Winters, "A Wireless Network for Wide-Band Indoor Communications", IEEE JSAC, June 1987.
16. Gfeller F.R. and Bapst U., Wireless in-house data communication via diffuse infrared radiation, IEEE Proceedings 67, 11 (1979), 1474-1486.
17. C.S. Yen and R.D. Crawford, "The Use of Directed Beams in Wireless Computer Communications," Proceedings of the IEEE Globecom, New Orleans, LA,Dec. 1985, pp. 1181-1184
18. J. R. Bary, J.M. Kahn, E.A. Lee and D.G. Messerschmitt, "Simulation of Multipath Impulse Response for Indoor Diffuse Optical Channel," Proceedings of the IEEE Workshop on Wireless LANs, Worcester, MA, May 1991, pp. 81-87.
19. K. Zhang and K. Pahlavan, "CSMA Local Radio Networks with BPSK Modulation in Rayleigh fading Channels", IEE Elect. Let., Sep. 27th,1990.
20. K. Zhang and K. Pahlavan, " Relation Between Transmission and Throughput of the Slotted ALOHA Local Packet Radio Networks"IEEE Trans. on Comm., March 1992.
21. K. Zhang and K. Pahlavan, " An Integrated Voice-Data System for Wireless Local Area Networks," IEEE Trans. on VT, April 1990.

Comparison Between Radio and IR Systems

Technique	DF/IR	DB/IR	RF	RF/SS
Data rate	1 Mbps	10 Mbps	15 Mbps	2-6 Mbps
Mobility	Good	Non	Better	Best
Detectability	Non	Non	Some	Little
Range	70-200 ft	80 ft	40-130 ft	100-800 ft
Frequency/Wavelength	λ=800-900 nm	λ=800-900 nm	f=18 GHz	f=0.9, 2.4,5.7 GHz
Radiated power	—	—	25 mW	< 1 W

Table I: Comparison between wireless LAN technologies

Comparison Between Mobile Data Services

Network	Modulation	Wireless Access	Data Rate	Provider
ARDIS	PSK	FDMA	4,800 bps	IBM/Motorola
Mobitex	GMSK	ALOHA	8000 bps	Ericsson/Swedish(RAM MObi)
CDPD	GMSK	ALOHA (over AMPS)	19,200 bps	IBM/Operating companies
IS-95	DSSS	CDMA (over mobile radio)	9600 bps	QUALCOMM

Table II: Comparison between mobile data networks

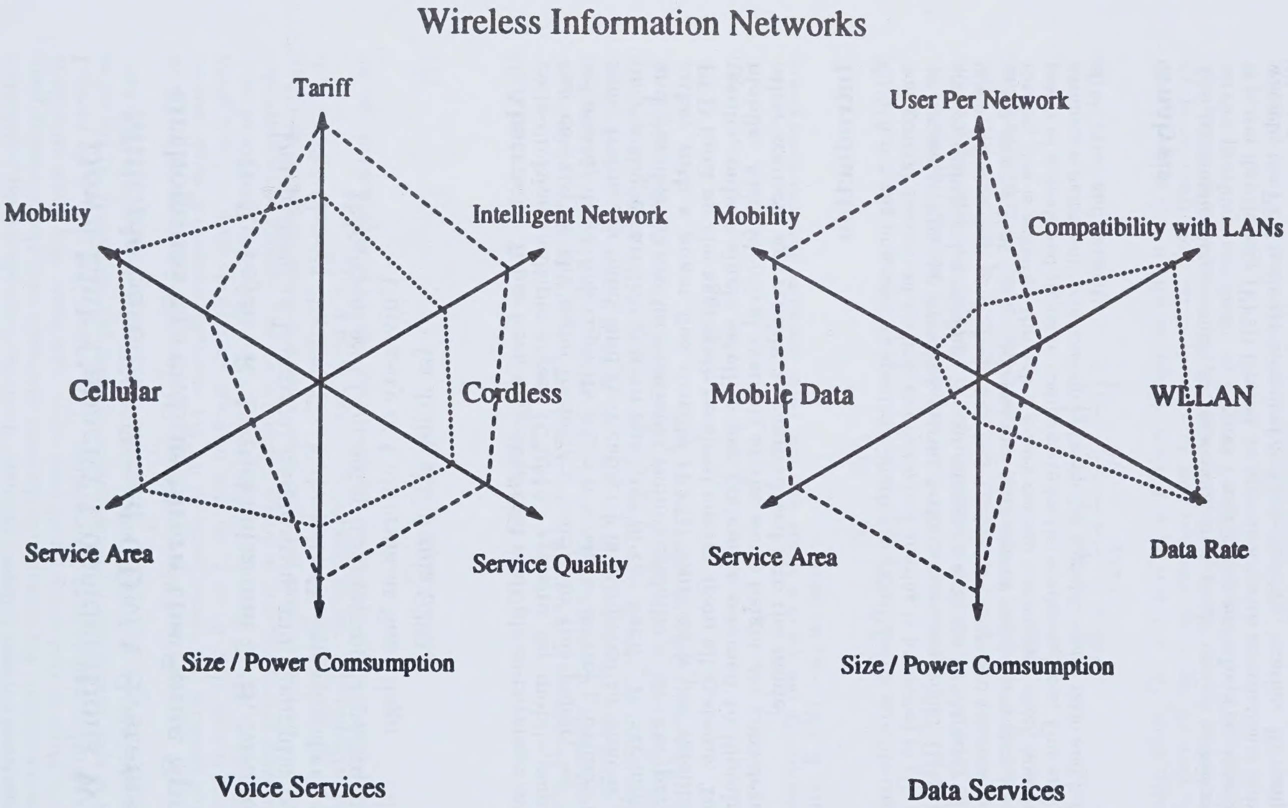


Figure 1: Various aspects of the voice and data services offered by wireless information network industry. Mobile cellular is compared with the cordless per-

Down-Link Capacity Considerations When Multiple Satellite Based CDMA Systems with Orthogonal Spreading Share the Same Spectrum

B. R. Vojcic, R. L. Pickholtz and L. B. Milstein[†]
Department of Electrical Engineering and Computer Science
The George Washington University Washington, D.C. 20052
[†]Electrical and Computer Engineering Department
University of California at San Diego
La Jolla, California 92093

Abstract - **It has been suggested that multiple co-coverage satellite-based code-division multiple access (CDMA) systems for mobile communications can co-exist in the same frequency band. In this paper, we quantify the achievable down-link capacity when n separate service providers all share the same frequency band, and that capacity is compared to what is achievable if only a single satellite system operates in the band. To account for current and possible future international radio regulations, the comparison is done either with a power flux density (PFD) limit on a per satellite basis or a PFD limit on the aggregate received energy from all systems. To get specific results, only a single satellite per system is assumed to illuminate a given mobile. Also, identical systems at the same height are considered, although other scenaria are readily accommodated into the model.**

Introduction

CDMA is being proposed for spectrum sharing in global mobile satellite communications, and general analysis of up-link and down-link sharing is presented in [1]. In this paper, we concentrate on the down-link sharing performance of multiple CDMA systems which employ orthogonal spreading. It is of interest to evaluate the capacity of such systems when different degrees of shadowing to terrestrial receivers are assumed.

We assume that both shadowed and unshadowed users experience the same performance; this is achieved through a power control mechanism which boosts the transmit power of a shadowed signal to compensate for the shadowing loss. Our analysis implicitly assumes a monotonic relationship between the signal-to-noise ratio and both probability of bit error and capacity.

Analysis

The communication scenaria corresponding to a single service provider and multiple service providers are shown in Figures 1 and 2, respectively. The down-link operation is power flux density (PFD) limited, in accordance with international radio regulations, whereby the PFD is always measured at a non-shadowed location. By comparing Figures 1 and 2, it is apparent that in the multiple service providers' case, each user has to handle multiple access interference (MAI) from "OTHER" systems, because orthogonality

---

[*] This work was supported by Motorola Satellite Communications.

cannot be maintained between different systems. However, there are several possibilities which must be considered. For example, in the most optimistic case, the processing gain, L, is sufficiently large so that every signal from every spot beam of a given satellite is orthogonal to all other signals from that satellite. This implies that $JK_1 \leq L$, where J is the number of spot beams and K_1 is the number of users per spot beam. At the other extreme, we might have $K_1 = L$, which implies that orthogonality only exists within each spot beam.

In addition, the comparison of the performance of n service providers with that of a single service provider has many facets to it. For example, suppose a single service provider system is such that $K_1 = L$, so that orthogonality exists only within a given spot beam. For an n service provider scenario, we assume that each satellite of each system has the same processing gain as does a satellite when only a single service provider exists. But K_n, the number of users per spot beam when n systems coexist, is less than the analogous quantity for one system, namely K_1. Therefore, if $K_1 = L$, then $K_n < L$ and additional orthogonality can be used to advantage when $n > 1$.

If we consider first a single system of satellites, then the PFD requirement can be expressed as [1]

$$K_1 P_1 I_0 \gamma \leq P_{max}, \tag{1}$$

where

$$\gamma = 1 - B + PB, \tag{2}$$

B is the percent shadowing in the system and P represents a power amplification factor designed to compensate for shadowing loss and fading (see [1,2]). In (1) and the equations to follow, P_n represents the transmit power per user when n service providers coexist, and P_{max} is the power constraint determined by the PFD limit. To gain a perspective on the effect of orthogonality, consider the following possibilities for a single system. If $JK_1 \leq L$, then, at a non-shadowed location, we have

$$SNR_1 = \frac{2E_1}{N_0}, \tag{3}$$

where E_1 is the energy per bit of a single user and N_0 is one-sided thermal noise power spectral density. If $JK_1 > L$, we have

$$SNR_1 = \left(\frac{N_0}{2E_1} + \frac{K_1 I_1}{2L}\gamma\right)^{-1}, \tag{4}$$

where $0 < I_1 < I_0$. The parameters I_0 ans I_1 re defined in [1]; basically, I_0 represents the ratio of the total MAI from J spot beams to the MAI within the spot beam of the user of interest, without orthogonal spreading, and I_1 is the ratio of the total MAI when orthogonal spreading is employed to the MAI within a spot beam without orthogonal spreading. When $K_1 = L$, then

$$SNR_1 = \left[\frac{N_0}{2E_1} + \frac{K_1(I_0 - 1)}{2L}\gamma\right]^{-1}. \tag{5}$$

More specifically, when $JK_1 > L$ and $K_1 < L$, then (4) applies with $0 < I_1 < I_0 - 1$, and when $K_1 > L$, then (4) again applies, but now $I_0 - 1 < I_1 < I_0$. Note that, for future reference, if we define $I_{1,\infty}$ to be the value of I_1 when $E_1/N_0 \to \infty$, then we obtain the maximum capacity per spot beam given by

$$K_{1,\infty} = \frac{2L}{I_{1,\infty}\gamma SNR_1}. \tag{6}$$

From (6), we see that $K_{1,\infty}$ depends upon the system design parameters L and SNR, and on the amount of shadowing in the systems which is reflected by γ. Also, $I_{1,\infty}$ is implicitly a function of $K_{1,\infty}$. For example, if we assume that $K_{1,\infty} > L$, then $I_{1,\infty} = I_0 - L/K_{1,\infty}$ and (6) reduces to

$$K_{1,\infty} = \frac{L}{I_0}\left(1 + \frac{2}{\gamma SNR_1}\right). \tag{7}$$

To ensure that $K_{1,\infty} > L$, from (7), we must have

$$\gamma < \frac{2}{SNR_1(I_0 - 1)}, \tag{8}$$

or, equivalently,

$$B < \frac{1}{P-1}\left[\frac{2}{SNR_1(I_0 - 1)} - 1\right], \tag{9}$$

which indicates that the percent shadowing must be small, as one would expect. On the other hand, if (8) is not satisfied, $K_{1,\infty} < L$ and can be bounded by

$$\frac{2L}{(I_0 - 1)\gamma SNR_1} < K_{1,\infty} < L. \tag{10}$$

We now consider n simultaneously operational systems, and define I_n, analogous to I_1, as the ratio of the total MAI from J spot beams of a single satellite, with orthogonal spreading, to the per spot beam MAI without orthogonal spreading. Note that because $K_n < K_1$, then $I_n < I_1$. The composite down-link signal-to-noise ratio is given by

$$SNR_n = \left[\frac{N_0}{2E_n} + \frac{K_n I_n}{2L}\gamma + (n-1)\frac{K_n I_0}{2L}\gamma\delta\right]^{-1}, \tag{11}$$

where

$$\delta = 1 - B + \frac{B}{c}, \tag{12}$$

and c represents the shadowing loss (note that $c > 1$).

If we have a PFD limit per system, then we have

$$K_n P_n I_0 \gamma \leq P_{max} \tag{13}$$

as the power constraint equation, and we obtain

$$\frac{nK_n}{K_1} = \frac{n}{1 + \frac{K_1}{K_{1,\infty}}\left\{\left[(n-1)I_0\delta + I_n\right]\frac{1}{I_{1,\infty}} - \frac{I_1}{I_{1,\infty}}\right\}}. \tag{14}$$

From (14), it can be shown that $nK_n > K_1$ if either of the following conditions are satisfied:

$$n > 1 - \frac{I_1 - I_n}{\frac{K_{1,\infty}I_{1,\infty}}{K_1} - I_0\delta}, \text{ for } \frac{K_1}{K_{1,\infty}} < \frac{I_{1,\infty}}{I_0\delta}, \tag{15}$$

or

$$n < 1 - \frac{I_1 - I_n}{\frac{K_{1,\infty}I_{1,\infty}}{K_1} - I_0\delta}, \text{ for } \frac{K_1}{K_{1,\infty}} > \frac{I_{1,\infty}}{I_0\delta}. \tag{16}$$

With an aggregate PFD limit defined by

$$nK_nP_nI_0\gamma \le P_{max}, \tag{17}$$

we obtain

$$\frac{nK_n}{K_1} = \left\{1 + \frac{K_1}{I_{1,\infty}K_{1,\infty}}\left\{\frac{1}{n}\left[(n-1)I_0\delta + I_n\right] - I_1\right\}\right\}^{-1}, \tag{18}$$

and a single service provider always yields a greater capacity as long as

$$I_1 < \left(1 - \frac{1}{n}\right)I_0\delta + \frac{I_n}{n}. \tag{19}$$

Note that (19) is satisfied for values of n defined by

$$n > \frac{I_0\delta - I_n}{I_0\delta - I_1}, \text{ for } \delta < \frac{I_1}{I_0}, \tag{20}$$

or

$$n < \frac{I_0\delta - I_n}{I_0\delta - I_1}, \text{ for } \delta > \frac{I_1}{I_0}. \tag{21}$$

Consider next specializing the above results to correspond to the various degrees of orthogonality described above. When the capacity per spot beam is small, such that $JK_1 \le L$, we have $I_1 = I_n = 0$. Then, it can be shown from (14) that n systems yield larger capacity than a single system when a PFD limit is on a per system basis. However, if an aggregate PFD limit is imposed, then maximum capacity occurs when $n = 1$, because (21) always applies.

Suppose now, for both the single system and multiple systems, orthogonality is only maintained within each spot beam of each satellite. Eq. (5) gives the signal-to-noise ratio for $n = 1$, given $K_1 \le L$, and SNR_n is given by

$$SNR_n = \left[\frac{N_0}{2E_n} + \frac{K_n(I_0 - 1)}{2L}\gamma + (n-1)\frac{K_nI_0}{2L}\gamma\delta\right]^{-1}. \tag{22}$$

If the PFD limit is on a per system basis, then it can be shown that n systems always outperform a single system if

$$K_1 < \frac{2L}{I_0\gamma\delta SNR_1}; \tag{23}$$

otherwise, a single system achieves maximum capacity.

For an aggregate PFD limit, if $\delta > 1 - 1/I_0$, then maximum capacity is achieved for $n = 1$. If the inequality is reversed, then $n > 1$ is optimal.

Numerical Results

To obtain specific results when orthogonal spreading is employed, we assume that a convolutional code of constraint length 9 and rate 1/2 is used. For such a code, a non-shadowed user needs $SNR_1 \ge 1.62$ to achieve $BER \le 10^{-3}$ [2]. Assuming a data rate per user of 4.8 kbps and a spread bandwidth of 1.25 MHz, the processing gain per coded symbol is $L = 130$. Assuming $J = 19$ spot beams per satellite, I_0 is defined by

$$I_0 = 1 + 6I_{0_1} + 12I_{0_2}. \tag{24}$$

Typical values of I_{0_1} and I_{0_2} are 0.15 and 0.01, respectively, so that $I_0 = 2.02$, assuming that the orthogonality is maintained only within each spot beam. I_{0_1} represents the ratio of the multiple access interference from an adjacent cell to the multiple access interference from the cell-of-interest. Similarly, I_{0_2} is defined as the ratio of multiple access interference from a cell in the second tier to the multiple access interference from the cell-of-interest.

Assuming $c = 10dB$ and $P = 12dB$, for a per system PFD limit, we present in Table 1 some characteristic values of system parameters when orthogonal sequences are employed.

Table 1: Characteristic values of system parameters with orthogonal spreading sequences.

B	0	0.095	0.31	0.62
γ	1	2.42	5.67	10.3
δ	1	0.91	0.72	0.44
$K_{1,\infty}/L$	1.12	1	0.5	0.33
$I_{1,\infty}$	1.44	1.02	0.87	0.72
$(K_1/K_{1,\infty})_{lim}$	0.71	0.55	0.59	0.81
$(K_1/L)_{lim}$	0.8	0.55	0.3	0.27

The fourth and fifth rows correspond to the limiting performance of a single system (when $E_1/N_0 \to \infty$), for a given amount of shadowing (B). The last two rows indicate the limiting values of capacity for a single system such that single system capacities less than these values result in n systems outperforming the single system. That is, any $n > 1$ achieves larger capacity than a single system when the capacity of that single system is smaller than the appropriate value given in the last two rows of Table 1.

The capacity due to multiple service providers is shown in Figures 3-4. In both figures, corresponding to orthogonality only within each spot beam, capacities are normalized with respect to the processing gain L (maximum number of orthogonal sequences). In Figure 3, n is taken to be 3, and a family of curves for different values of B shows the resulting capacity. Curves are plotted only up to the point where $n = 3$ yields larger capacity than $n = 1$. In Figure 4, for the value of $B = 0.2$, the capacity is shown versus K_1/L for various values of n.

When the orthogonality is fully employed both by a single system and by multiple systems beyond just on a per spot beam basis (i.e., if $L > K_1$, then the orthogonality extends to more than one spot beam per set of orthogonal signals), and when $K_1/K_{1,\infty} < I_{1,\infty}/(I_0\delta)$, (15) indicates that n systems ($n > 1$) always yields larger capacity than does a single system. When the above inequality is reversed (i.e., that $K_1/K_{1,\infty} > I_{1,\infty}/I_0\delta$), it can be shown from (16) that for most realistic sets of parameters values $n = 1$ is optimum.

Finally, we consider the case in which orthogonal sequences are employed on the downlink and an aggregate PFD limit is imposed. When orthogonality is maintained only within each spot beam, then, if $\delta < (1 - 1/I_0)$, n systems yield larger capacity than does a single system (see (20), and note that for this case $I_1 = I_n = 1 - 1/I_0$). If the inequality is reversed, $n = 1$ is optimal. Figures 5 and 6 compare the capacity of n systems with the capacity of a single system for $B = 0.2$ and $B = 0.6$, corresponding to capacity degradation and capacity improvement, respectively.

Conclusions

Whether a single system or multiple systems achieve greater capacity is a function of a variety of parameters, among which are the degree of shadowing and the type of PFD limit.

It has been shown in this paper that under a PFD limit per satellite, multiple systems can achieve a larger total capacity than a single system only if the capacity of the single system is smaller than some value, which depends upon the imposed PFD limit and the amount of shadowing in the system. As the percentage of shadowing increases, the crossover point becomes smaller and multiple systems can achieve a capacity improvement only in a relatively low capacity region. This is a consequence of the near-far problem induced by the interaction of shadowing and the power control mechanism introduced to compensate for it. On the other hand, when an aggregate PFD limit is imposed on the operating systems, we find that a single system always achieves a larger capacity than the total of n systems if the amount of shadowing is relatively low. When the shadowing in the system becomes excessive, the opposite conclusion is true. In that case, shadowing actually helps by reducing the total multiple access interference as seen by each mobile user (on the average). This leads to smaller transmit power per user, and enables more users to operate within the PFD constraint.

Acknowledgment

We would like to acknowledge the help and motivation provided to us by Dr. Ray Leopold and Dr. Carrie Devieux of Motorola Satellite Communications.

References

[1] B.R. Vojcic, R.L. Pickholtz and L.B. Milstein, "Total Capacity in a Shared CDMA LEOS Environment", submitted to the IEEE Journal on Sel. Areas in Communications.

[2] B.R. Vojcic, L.B. Milstein and R.L. Pickholtz, "Power Control Versus Capacity of a CDMA System Operating Over a Low Earth Orbiting Satellite Link". To be presented at the Communication Theory Mini-Conference in conjunction with Globecom'93.

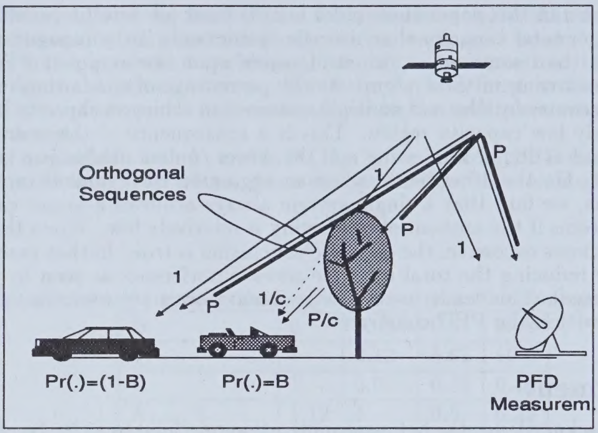

Orthogonal
Sequences
1
P
P
1
1
P
1/c
P/c
Pr(.)=(1-B)
Pr(.)=B
PFD
Measurem.

Figure 1: Down-link shadowing scenario with orthogonal spreading; single service provider.

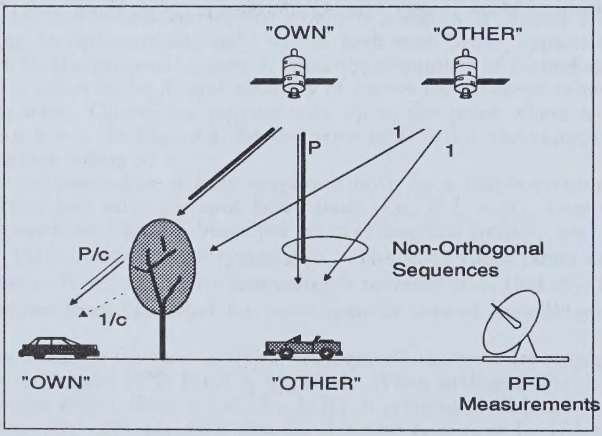

"OWN"
"OTHER"
1
P
1
P/c
1/c
Non-Orthogonal
Sequences
"OWN"
"OTHER"
PFD
Measurements

Figure 2: Down-link shadowing scenario with orthogonal spreading; multiple service providers.

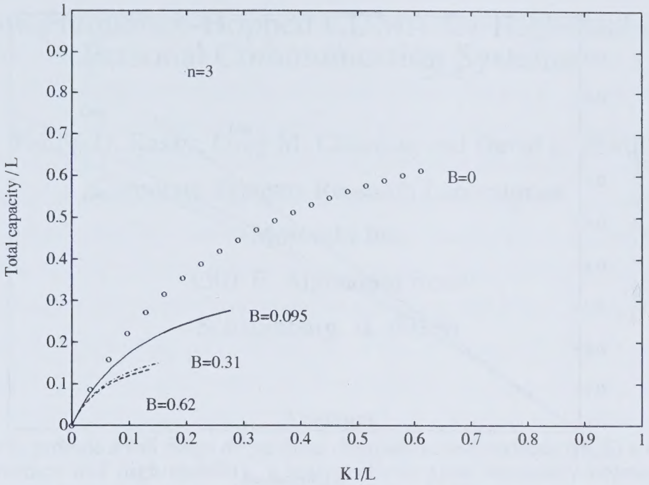


Figure 3: Down-link capacity of 3 satellites when orthogonality is employed just on a per spot-beam basis; PFD limit per satellite.

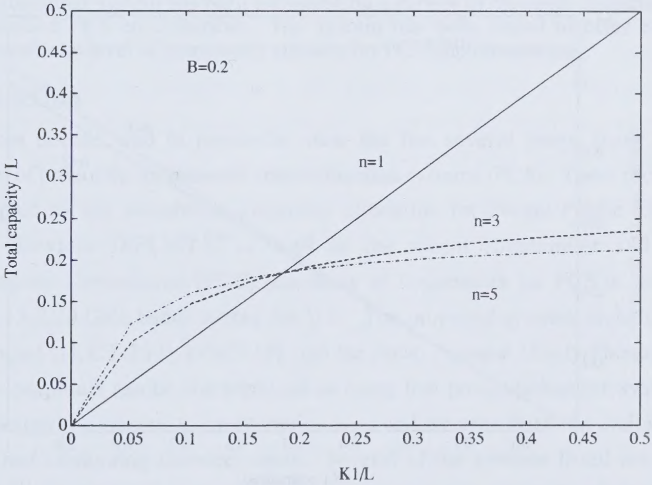


Figure 4: Down-link capacity of n satellites when orthogonality is employed just on a per spot-beam basis; PFD limit per satellite.

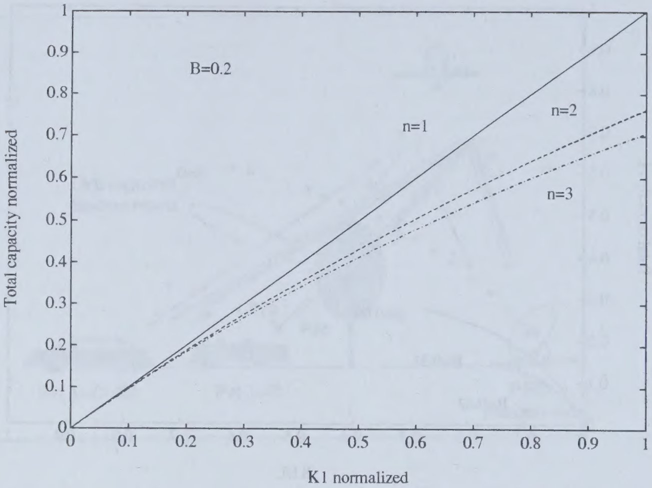


Figure 5: Down-link capacity of n satellites when orthogonality is employed just on a per spot-beam basis; aggregate PFD limit (B=0.2).

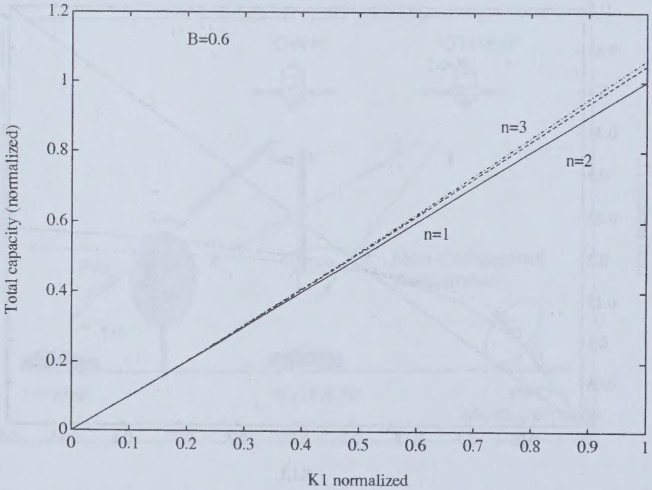


Figure 6: Down-link capacity of n satellites when orthogonality is employed just on a per spot-beam basis; aggregate PFD limit (B=0.6).

Slow Frequency-Hopped CDMA for High-Mobility Personal Communication Systems

Phillip D. Rasky, Greg M. Chiasson and David E. Borth
Corporate Systems Research Laboratories

Motorola Inc.

1301 E. Algonquin Road
Schaumburg, IL 60196

Abstract

In order to provide a full range of personal communication services (PCS) with wide-area coverage and high mobility, a macrocellular slow frequency-hopped (SFH) CDMA complement to microcellular PCS is defined. This system was designed to permit true vehicular/portable mobility with larger cells than conventional PCS proposals would allow while providing for the existence of a lower-cost PCS which would be operable in truly pedestrian-only environments. This SFH CDMA system employs the multiple access technique of slow frequency hopping to provide robust transmission over the harsh mobile radio channel and to maintain high capacity despite the use of larger cells. This paper discusses the alternatives considered in the definition of this system, highlighting choices which allow significant reductions in complexity without compromising system performance. In addition, the capacity of the macrocellular system has been estimated on a variety of channels characteristic of the expanded PCS environment. The system has been found to offer excellent performance at a level of complexity suitable for PCS implementation.

1 Introduction

Over the last decade, and in particular, over the last several years, there has been a proliferation of proposals for personal communication systems (PCS). These proposals have been prompted by the worldwide frequency allocation for Future Public Land Mobile Telephone Services (FPLMTS) as well as the recent (September 1993) Federal Communications Commission (FCC) allocation of frequencies for PCS in the 1.85-1.97 GHz and 2.13-2.20 GHz bands within the U.S. The proposed systems include Bellcore's UDPC proposal [1], CT-2 [2], DECT [3], and the Japan Personal Handy Phone system [4]. All of these proposals can be characterized as being low-power pedestrian systems which have been designed to operate in small service areas and are expected to be available for both low initial and continuing (service) costs. Several of the systems listed are capable of handoff between service areas, or cells; however, in each case, the cell radius extends to at most 1000-2000 feet usually due to transmit power and antenna height limitations. Providing continuous coverage in a large metropolitan area would not appear to be economically feasible with such small cells, except in densely populated regions. Furthermore, providing vehicular mobility would seem to be impractical in such systems due to the extreme rate of handoffs.

To provide a full range of personal communication services with wide-area coverage and high mobility, it is desirable to define a macrocellular system to complement microcellular PCS. At first glance, merely increasing the antenna height or the transmitter power levels would seem to permit air interfaces designed for microcellular deployment to be suitable for the vehicular environment. However, a high-mobility macrocellular communication system must simultaneously compensate for the effects of delay-spread channels characteristic of higher base antenna heights and the reduction in overall system capacity relative to microcellular PCS. In this paper, a system which overcomes these problems is described. This system was designed to permit true vehicular/portable mobility with larger cells than so-called "conventional" PCS systems would allow while remaining true to the fundamental PCS tenets of low cost and low complexity. Furthermore, the system provides a flexible backbone to support emerging telecommunications services through the exploitation of the concept of "bandwidth on demand."

2 System Overview

In order to determine the best method of meeting the goals established for high mobility/wide area PCS, a simulation study was conducted which examined different combinations of options for a mobile communication system with a 500 kbps gross bit rate on a time-division multiple-access (TDMA) carrier providing up to 10 voice channels or, through the concatenation of multiple time slots, data services with net data rates up to 144 kbps. The options examined included possible combinations of: (1) channel coding and interleaving; (2) diversity combining; (3) slow frequency hopping (SFH); and (4) channel equalization. The goal of this simulation study was to determine the combination of these options which would yield the lowest carrier-to-interference (C/I) protection ratio requirement which, in turn, translates into the highest system capacity.

For the purposes of the comparisons made in this section, a linear receiver was employed, and the selection of the channel coding technique was limited to rate $\frac{1}{2}$ convolutional codes with soft-decision decoding and interleaving spanning 40 msec to limit the voice channel delay. In each case, the modulation technique was QPSK with burst-coherent demodulation and carrier recovery from a preamble inserted in each TDMA slot. In addition, a source bit rate of 32 kbps to support industry-standard ADPCM was assumed. This requires two out of the total of 10 timeslots per TDMA frame. In the receiver processing, two branch maximal ratio diversity combining was employed in the systems incorporating diversity, and a sixteen state maximum likelihood sequence estimation (MLSE) equalizer was used for the equalized results. Interference was modeled as a faded pseudo-random data source with modulation identical to that of the desired signal. Base stations were assumed to be synchronized to the extent that corresponding TDMA slots of adjacent sites were aligned.

The performance of the candidate systems has been evaluated on several slowly fading (5 mph) channels which characterize the range of conditions expected in a macrocellular PCS environment. A non-frequency selective Rayleigh-fading channel and three delay-spread channels were considered. For the case of frequency-hopped systems, it was assumed that the system bandwidth was much greater than the coherence bandwidth, therefore resulting in independent fading from hop-to-hop [5]. The first delay-spread channel was based upon the COST 207 typical urban profile [6] used in the validation testing of the Pan-European digital cellular system, GSM. The second and third delay-spread channel models were two-ray, equal-gain models with one and two symbols of differential delay (4 and 8 μsec), respectively. For each of the delay-spread channel models, all of the paths were assumed to undergo independent Rayleigh fading with a power spectrum determined by the classical (Jakes) model for Rayleigh fading in the land mobile radio environment.

Tables I and II summarize the results of this study by enumerating the C/I ratio required to achieve a decoded bit error rate (BER) of 10^{-3} for various combinations of the options. The results presented were obtained using a constraint length six convolutional code. The performance of a constraint length nine code was also evaluated; however, the increased constraint length generally resulted in a performance gain of less than 1 dB while adding significantly more complexity to the system. In reviewing the results, it is apparent that the combination of channel coding, antenna diversity combining, and slow frequency hopping best meets the defined goals.

The combination of diversity and forward error correction coding allows robust reception on the harsh mobile radio channel, particularly in multipath conditions. Frequency hopping provides a form of frequency diversity to the system and, by effectively decorrelating the fading process, reduces the duration of long fades [7]. This results in good performance at low speeds and in general, performance which is independent of speed. Within a cell, the frequency-hop patterns are orthogonal; therefore, there is no intracell interference in the SFH system. Since the orthogonality is based upon frequency division, it is maintained on both the up and downlinks and in the presence of multipath channel conditions and propagation delays. This is a significant advantage over alternative code division multiple access (CDMA) proposals and can result in a sizable capacity increase. Furthermore, the intercell co-channel interference is sampled rather than averaged. By virtue of the frequency-hopping pattern, a given source of interference can affect the system only a small portion of the time. Therefore, the near/far problem is eliminated, and there is no danger of a single rogue user disrupting an entire cell site. (In [8] and [9] this is referred to as "interferer diversity.") Taken together, these benefits allow the use of single cell reuse patterns with high loading factors thereby enabling the macrocellular system to meet the capacity requirements demanded of a PCS.

Comparing the results of Table I and Table II, it is interesting to note that given the two system attributes of diversity and frequency hopping, the system does not require an equalizer in multipath of up to several microseconds of RMS delay spread. The equalizer improves performance only slightly on the more severe channels and actually degrades

Table I - C/I (dB) required for 10^{-3} decoded BER without MLSE equalization.

	No Diversity		Diversity	
	Non-hopped	Frequency Hop	Non-hopped	Frequency Hop
Rayleigh Fading	18.0	6.1	6.8	1.2
Typical Urban	17.4	7.4	7.5	1.8
1 Symbol Delay	15.0	9.9	7.5	3.6
2 Symbol Delay	error floor (>20)	13.0	8.6	4.7

Table II - C/I (dB) required for 10^{-3} decoded BER with MLSE equalization.

	No Diversity		Diversity	
	Non-hopped	Frequency Hop	Non-hopped	Frequency Hop
Rayleigh Fading	19.0	8.2	8.6	3.2
Typical Urban	16.7	8.6	9.0	2.9
1 Symbol Delay	13.1	8.1	6.8	3.4
2 Symbol Delay	13.1	8.1	7.2	3.5

performance on the Rayleigh fading channel and typical urban profile (approximately 1 μsec RMS delay spread). This degradation can be attributed to noise enhancement of the received signal by the equalizer at very low signal-to-noise ratios.

3 Hardware Description

In keeping with the fundamental PCS requirement of low cost communications, great efforts have been taken to minimize the complexity of the transceiver. In particular, the elimination of an equalizer presents a significant opportunity to reduce the complexity of the slow frequency-hopping receiver. Typically, an adaptive equalizer in mobile radio environment requires a linear, wide dynamic range front-end. With this restriction removed, it is possible to employ a hard-limiting receiver. Due to the hard-limiting receiver, it is possible for the personal communicator to synchronize to the system much faster as receiver gain is removed as a variable from the synchronization process. Furthermore, the elimination of the signal processing complexity of the adaptive equalizer and the linear front-end circuitry, automatic gain control (AGC), and wide-dynamic range A/D conversion, all associated with linear receivers, significantly reduces the complexity impact of the dual receiver branches. Link simulations have shown that the loss from the hard-limiting front-end is typically less than 1 dB in C/I protection. This is a significant development as hard-limited receivers are not usually associated with receivers striving for soft-decision decoding and diversity combining gains.

A complete duplex link prototype with the characteristics shown in Table III has been developed at Motorola and is described in detail in [10]. The PCS transceiver resulting from these design considerations is depicted in high-level block diagram form in Figure 1. The prototype was designed to validate field performance of low complexity approaches to wide-area coverage, high-mobility personal communications. The modulation format, data rate, TDMA structure, slot timing, and frame timing of the prototype system are all compatible with the proposed Bellcore UDPC system. As Figure 1 indicates, the transceiver may be logically separated into transmitter and receiver functions. The transmitter consists of a speech coder followed by a channel coder and interleaver. The encoded data is four-phase modulated, and frequency hop transmission is used to help mitigate channel impediments such as slow fading.

Table III - Attributes of the SFH PCS System.

Frequency band:	1850-1990 MHz
Slots/frame	10
Channel bit rate	500 kbps
Frequency hopping rate	500 Hz
Duplex method	FDD
Intracell multiplexing	TDM/FDM (orthogonal SFH)
Intercell multiplexing	SFH/CDMA
Modulation	QPSK
Pulse Shaping	raised cosine, a=0.5
Time slot duration	0.2 msec
Frame duration	2.0 msec
Gross data rate/slot	34 kbps
Channel coding:	
Type	Rate 1/2, K=6 Convolutional, soft decision
Interleaving span	40 msec
Maximum differential multipath delay	16 µsec
Speech coder	32 kbps ADPCM

The receiver portion of the transceiver is designed to efficiently detect and decode the transmitted signal. In view of the fading and multipath channels which are prevalent in mobile communications, a diversity receiver is employed to improve performance. At the receiver, an RF front end tracks and downconverts the hopping signal for each branch. Within this conversion process, the signal is hard limited which results in significant cost and complexity savings. Following the front end, the proper sampling point for each branch is determined, and the signals are combined in a method approximating max-ratio diversity. The combined signal is deinterleaved, and soft decision Viterbi decoding is performed to recover the original data which is output through the speech decoder.

The mobile and base station hardware is assembled in a 19" rack with modules that are 10" high and 12" deep. The transceiver hardware is suitably constructed for vehicular field testing. The base and mobile units can transmit a peak power of 800 and 500 mW, respectively. The system is fully capable of establishing both mobile originated and mobile terminated telephone calls with an interface to the PSTN through an echo-cancelled conventional analog line. Considerable special purpose testing software has been integrated into the units to allow both objective and subjective quality assessments of the full-duplex link. The experimental hardware is capable of logging error rate statistics, received signal strength, and carrier-to-interference ratios. More detailed statistics are available at test points throughout the experimental transceiver.

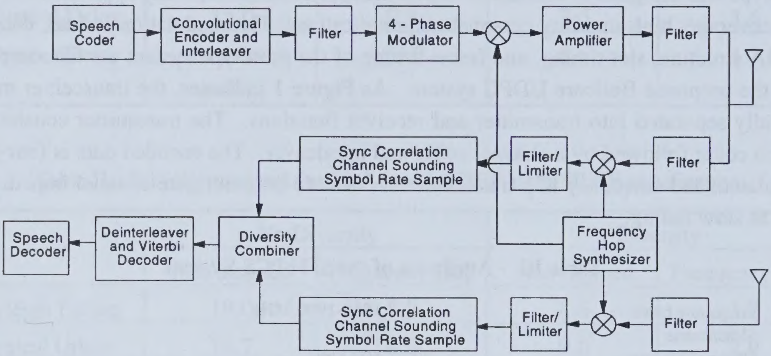

Speech Coder
Convolutional Encoder and Interleaver
Filter
4 - Phase Modulator
Power Amplifier
Filter
Sync Correlation Channel Sounding Symbol Rate Sample
Filter/ Limiter
Filter
Speech Decoder
Deinterleaver and Viterbi Decoder
Diversity Combiner
Frequency Hop Synthesizer
Sync Correlation Channel Sounding Symbol Rate Sample
Filter/ Limiter
Filter

Figure 1. SFH/TDMA PCS transceiver high-level block diagram.

Key to the performance of such a receiver is the proper calculation of the weighting coefficients which serve the dual purpose of providing confidence weighting for diversity combining and soft-decision decoding. Particular care must be taken when employing a hard-limited receiver, especially when the SFH system is used for code-division multiple access. Conventional practice employs a scaling based upon a measure of the received signal strength or channel gain. However, in SFH systems, the received signal is characterized by significant amplitude variation from hop-to-hop due to the effect of decorrelation of the fading process by hopping over a bandwidth greater than the coherence bandwidth of the channel. Furthermore, in the multiple access environment, where the interference level is likely to rapidly vary as the interference originates from a different source from hop-to-hop, the relative level of the interference must be taken into account. For use in the hybrid SFH/TDMA receiver, a weighting coefficient which yields significantly improved performance while allowing easy calculation has been derived. This coefficient takes the form of the ratio of the estimated channel gain ρ to the interference power $\sigma_n{}^2$ and may be calculated using only the synchronization word cross correlation R_{rx} and the received signal power after hard limiting $\sigma_r{}^2$ - two quantities which are readily available:

$$\frac{\rho}{\sigma_n^2} = \frac{R_{rx}}{\sigma_r^2 - R_{rx}{}^2} .$$

The limiter is often modeled as a circuit that normalizes the received waveform by the magnitude of the total received signal, desired signal plus interference. The hard-limiting process has removed the absolute gain information from the correlator output R_{rx}; however, the relative relationship between the channel gain and the interference power remains. Figure 2 shows a scatter plot of instantaneous (single hop) C/I estimates by the receiver as a function of the actual instantaneous C/I for a faded desired and faded interfering signal. Also shown is a polynomial curve fit to the data. The estimate based on the above equation is quite consistent with the actual carrier-to-interference ratio over the critical operating range of the receiver. The curve reaches a plateau of approximately 20 dB C/I due to the effects of quantization and finite precision arithmetic in the prototype receiver referenced above.

In the design of the experimental system, great efforts were made to develop a low cost, low complexity system. However, the required trade-offs were carefully balanced to have a minimal impact on performance. For example, at a 10^{-3} decoded bit error rate (BER), the loss of the hard limited, fixed point receiver relative to that of a linear, infinite precision receiver is approximately 1.2 dB. Thus, the goal of designing a high performance receiver for PCS has been achieved.

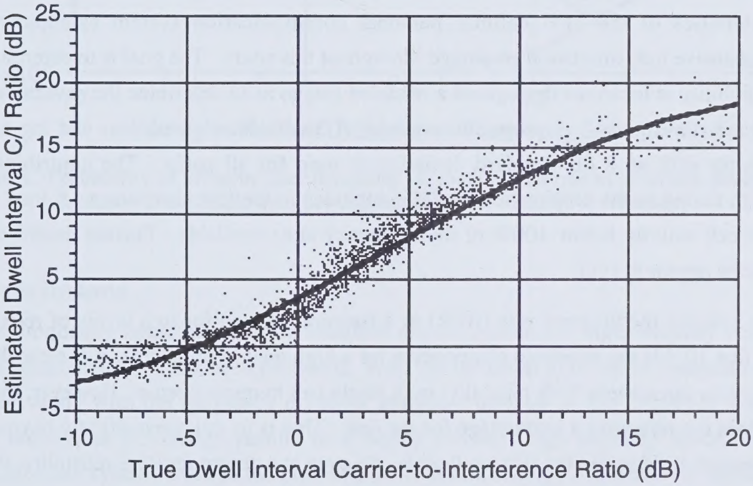


Figure 2. Estimated vs. actual C/I ratio on an ideally frequency selective fading channel at a mean C/I level of 2 dB.

4 Capacity

In addition to the performance measures described thus far, the capacity of such a hybrid SFH/TDMA system has been estimated. As mentioned above, slow frequency hopping has an inherent advantage as a CDMA technology since it is easy to maintain orthogonality between users of the same cell for both the uplink and downlink transmissions in spite of multipath channel conditions and propagation delay. Relative to DS/CDMA, the SFH uplink has no disadvantage in reliability or capacity due to intracell interference. In terms of intercell interference, however, the downlink for all CDMA systems applied to cellular-type communications is deficient, relative to the uplink, in its ability to exploit the concept of interference diversity or interference averaging to gain greater statistical reliability.

To produce an interference diversity effect, the concept of "interference averaging" relies on the premise that the interference experienced by any user is not subject to worst-case interference from a single, statistically dominant interferer, but rather from the aggregate of all the multiple-access users. Given a large number of users, this is true for the CDMA uplink as each potential interfering propagation path has independent path loss, shadowing, and fading characteristics. On the downlink, however, the interfering sources are relatively few in number compared to the uplink. They are the few base sites in closest proximity to a PCS unit. All the downlink transmissions from each of those base sites will interfere with the intended PCS unit over the same propagation path. Because of this, the downlink of the proposed SFH system limits capacity. As a result, the aspects of the study presented here are limited to downlink capacity.

A Monte-Carlo propagation simulation has been used to generate the interference characteristics of the SFH cellular personal communication system as input to the comprehensive link simulation employed throughout this study. The goal is to determine the link reliability at locations throughout a modeled cell so as to determine the severity of any cell fringe conditions. A propagation model of 36 dB/decade path loss and log-normal shadowing with an 8 dB standard deviation is used for all paths. The contribution of Rayleigh fading to the propagation model is included in the link simulation. A load factor models cell activity below 100% of the frequency slots available. Further details of this simulation appear in [11].

Figure 3 shows the bit-error rate (BER) as a function of load for two levels of reliability. Given that 10^{-3} is the threshold of operation for a high quality voice link, it is clear that the SFH system can achieve 95% reliability with single-cell frequency reuse. However, the 10^{-3} BER does not represent a true outage for the link. That is to say, even slightly beyond this point, speech is degraded but still intelligible. So even at a design for 95% reliability, there is still margin.

Assuming a sufficient number of carriers and 16 kbps speech coding technology, this suggests that each cell-site can support 10 one-way voice channels per 400 kHz of spectrum allocation or equivalently 10 two-way voice channels per 800 kHz of paired spectrum. The combination of the interference diversity effect of SFH/CDMA and the exploitation of voice inactivity in the air interface can yield as much as an additional factor of 2 in capacity, or equivalently 20 voice channels per 3-sector cell per 400 kHz of spectrum allocation. Alternatively, any additional gain due to the exploitation of voice inactivity can support higher reliability and better performance for the average user at a cell-site capacity of 10 voice channels per 400 kHz of spectrum allocation. In order to achieve voice-channel capacity in excess of 10 channels per 400 kHz, frequency reuse must occur more than once per sectored cell. While these capacity estimates assume a decoded BER of 10^{-3}, preliminary indications are that additional voice channel loading can be supported as modern speech coder designs can tolerate a BER greater than 10^{-3}, given a proper strategy for speech frame error detection and mitigation.

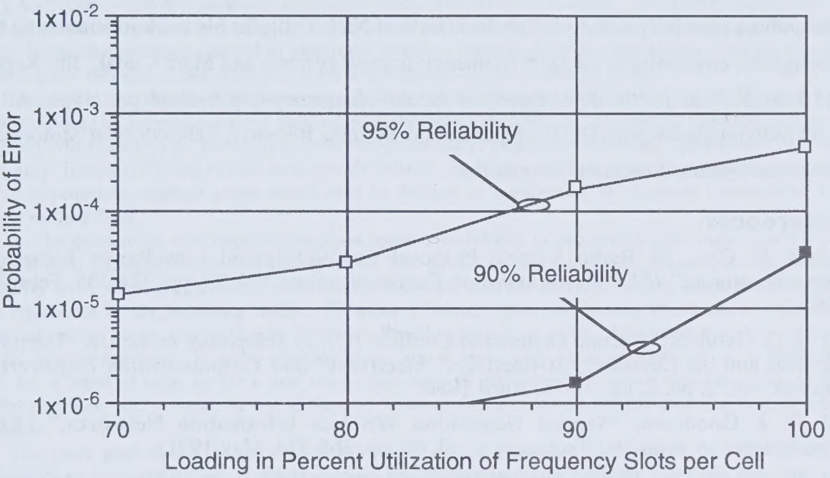


Figure 3. Probability of bit error after decoding vs. loading in terms of channels occupied vs. channels available on a per-cell basis for 95% and 90% reliability.

5 Conclusions

A slow frequency-hop/code-division multiple access system for high-mobility personal communication systems has been proposed. With the inclusion of diversity reception in the PCS unit along with the frequency diversity attributes inherent to the frequency-hopping access technique, the design results in a highly reliable, high capacity macrocellular communication system, even in the presence of multipath fading. Furthermore, this can be achieved without excessive amounts of complexity.

It has also been shown that through this hybrid access technique, single-cell frequency reuse is realizable. This is most easily achieved by a frequency mapping assigning orthogonal frequency sets to adjacent sectors in a 3-sector cell. Though this is not a requirement of the system, it seems only reasonable that as a logical extension to orthogonal transmissions within a sector, the sources of greatest potential interference be coordinated so as to exploit the propagation loss in wireless communications as an advantage.

The system concept is further enhanced by the attributes of a time-division multiple access carrier. Through the hybrid of TDMA and CDMA, bandwidth on demand can be delivered to support emerging telecommunications services for data, video, and high quality audio transmission. If PCS is to thrive in the marketplace, it must distinguish itself on the basis of high quality voice channels and services not provided by present-day wireless communication services.

Acknowledgments

The authors gratefully acknowledge the efforts of Nick Tolli, for his work in simulating the propagation environment for slow frequency-hopped systems and Mark Cudak, Jim Kepler and Steve Kuffner for the development of the slow frequency-hop research prototype. All of these individuals are with the Communication Systems Research Laboratory at Motorola's Corporate Systems Research Laboratories.

References

[1] D. C. Cox, "A Radio System Proposal for Widespread Low-Power Tetherless Communications," *IEEE Transactions on Communications*, vol. 39, pp. 324-335, February 1991.

[2] J. G. Gardiner, "Second Generation Cordless (CT-2) Telephony in the UK: Telepoint Services and the Common Air-Interface," *Electronics and Communication Engineering Journal*, vol. 2, no. 2, pp. 71-78, April 1990.

[3] D. J. Goodman, "Second Generation Wireless Information Networks," *IEEE Transactions on Vehicular Technology,* vol. 40, pp. 366-374, May 1991.

[4] K. Tanaka, M. Hirono, and I. Horikawa, "Signaling Architecture for Microcell Communication Systems," *Proceedings of the 41st IEEE Vehicular Technology Conference*, pp. 240-244, May 1991.

[5] W. C. Jakes (ed.), *Microwave Mobile Communications*, New York: John Wiley & Sons, 1974.

[6] COST 207 WG1, "Proposal on Channel Transfer Functions to be Used in GSM Tests Late 1986," COST 207 TD (86)51 Rev. 3, Paris, September 29-30, 1986.

[7] M. Mizuno, "Randomization Effect of Errors by Means of Frequency-Hopping Techniques in a Fading Channel," *IEEE Transactions on Communications*, vol. COM-30, pp. 1052-1056, May 1982.

[8] D. Verhulst, M. Mouly, and J. Szpirglas, "Slow Frequency Hopping Multiple Access for Digital Cellular Radiotelephone," *IEEE Transactions on Vehicular Technology,* vol. VT-33, pp. 179-190, August 1984.

[9] J. L. Dornstetter and D. Verhulst, "Cellular Efficiency with Slow Frequency Hopping: Analysis of the Digital SFH900 Mobile System," *IEEE Journal on Selected Areas in Communications*, vol. SAC-5, pp. 835-848, June 1987.

[10] P. D. Rasky, G. M. Chiasson, and D. E. Borth, "An Experimental Slow Frequency-Hopped Personal Communication System for the Proposed U.S. 1850-1990 MHz Band," *The Second International Conference on Universal Personal Communications*, Paper 44.03, October 12-15, 1993, Ottawa, Canada.

[11] P. D. Rasky, G. M. Chiasson, and D. E. Borth, "Hybrid Slow Frequency-Hop/CDMA-TDMA as a Solution for High-Mobility, Wide Area Personal Communications," *Fourth WINLAB Workshop on Third Generation Wireless Information Networks*, October 19-20, 1993, East Brunswick, NJ.

NONBLOCKING MULTI-CHANNEL SWITCHING

H. Saidi, P.S. Min - Washington University in St. Louis
M.V. Hegde - Louisiana State University

1 Introduction

An important issue for ATM is the development of a switching methodology that can support the very high bandwidth transmission associated with light wave technology. The transmission rate over optical fibers has reached tens of gigabits per second and is likely to increase in the coming years. On the other hand, the processing speed of electronic switches remains substantially slower. Such a mismatch between the two quantities creates a natural bottleneck at the switching nodes.

Multi-channel switches have been proposed and studied in the literature [1] [3] [6] [7], as a means of alleviating the processing speed constraint of electronic switches. Multi-channel switches can provide higher performance (e.g., throughput, cell loss probability, delay) by exploiting the concept of *channel grouping.* Instead of being routed to a specific output channel, a cell is routed to any channel belonging to an appropriate channel group which may be defined as a collection of channels transmitted over a single optical fiber.

As the demand for new applications soars, greater variability in bandwidth and traffic characteristics (e.g., session duration, burstiness) is expected. The advantages of statistically sharing a higher channel capacity under such conditions are well known. For example, the link efficiency is increased by reducing the burstiness in the incoming traffic. Bit pipes of higher rates are formed which allow a number of applications to share bandwidth by dynamically allocating time slots. Assuming that the exogenous traffic intensity is fixed, a larger channel group size is less likely to incur blocking for a single ATM cell, for a burst of cells, or for a new connection request. Other performance measures such as delay, buffer overflow, and congestion would improve when multiple channels are grouped together as a single resource.

The main goal of this paper is to develop a switching paradigm that accomplishes multi-channel switching as a generalization of single-channel switching paradigms that are currently prevalent. The resulting architecture must maintain the required features of single-channel switching (such as multicasting, and simplicity in hardware) while providing the advantages of multi-channel switching (such as multi-rate and super-rate switching, and higher performance).

In multi-channel switching, since cells belonging to a session can traverse multiple channels, and thus they can experience a widely differing amount of delay and congestion, maintaining the ordering among them becomes a challenging issue. An important goal of this paper is to develop a multi-channel architecture that guarantees the in-order cell sequencing without complex re-sequencing mechanism.

2 Definitions

With the definitions of the Flip Network (or equivalently the Inverse Omega Network) given in [5], we present the following definitions:

Definition 2.1 A *connection request* $r(a,b)$ where $a \in \{0,1\}^n$ and $b \in \{0,1\}^n$ represents the requirement to provide a set of edges in the network forming a path from $(a,0)$ to (b,n).

Definition 2.2 A *group connection request* $r(A,B)$ denotes the requirement to provide a set of edges in the network forming paths from each element a in set A to some element b in set B so that no element in B is reached more than once. This necessarily implies that $||A|| \leq ||B||$.

Definition 2.3 A network is said to be *internally nonblocking in the single-channel sense for a permutation* π *over* $\{0,1,2,\ldots,N-1\}$, if connection requests in the form of $\{r(i,\pi(i))\}_{i=0}^{N-1}$ are simultaneously supported by the network. A network, which is internally nonblocking in the single-channel sense for all permutations, is simply said to be *internally nonblocking in the single-channel sense.*

Definition 2.4 Let $Y_i \cap Y_j = \phi$ and $X_i \cap X_j = \phi$ for all $i \neq j$. A network is said to be *internally nonblocking in the multi-channel sense for* $(Y_0, Y_1, Y_2, ..., Y_{t-1}; X_0, X_1, X_2, ..., X_{t-1})$, if the set of group connection requests in the form of $\{r(Y_i, X_i)\}_{i=0}^{t-1}$ such that $\sum_{i=0}^{t-1} ||X_i|| \leq N$ and $||Y_i|| \leq ||X_i||$ for $i = 0,1,2,\ldots,t-1$ are simultaneously supported by the network. (Here t is an arbitrary integer such that $0 < t \leq N$.) A network is simply said to be *internally nonblocking in the multi-channel sense* if it is internally nonblocking in the multi-channel sense for any set of group connection requests.

Definition 2.5 Consider two cells appearing at the same time slot but at different input channels, $(a, 0)$ and $(\tilde{a}, 0)$, such that $a < \tilde{a}$. The cell appearing at the lower indexed channel (i.e., $(a, 0)$) is said to have the *higher relative ordering*. Similar relations can be defined at the output.

Definition 2.6 Consider two cells belonging to the same session. The cell generated earlier in time is said to have the *higher time ordering* than the other.

3 A Nonblocking Condition in Flip Networks

This section presents a particular nonblocking condition for the Flip network. This condition, which is stated as the Nonblocking Binary Group Lemma will be exploited in Section 4 to realize an efficient multi-channel switch that is internally nonblocking in the multi-channel sense.

Lemma 3.1 (Nonblocking Binary Group) Let $0 \leq K \leq N$. Let two sets Y and $\bar{Y}$ be defined such that $Y = \{y_0, y_1, y_2, \ldots, y_{K-2}, y_{K-1}\}$ and $\bar{Y} = \{\bar{y}_0, \bar{y}_1, \bar{y}_2, \ldots, \bar{y}_{N-K-2}, \bar{y}_{N-K-1}\}$ with the following constraints:

- (1) Monotone: $y_0 < y_1 < y_2 < \ldots < y_{K-1}$ *and* $\bar{y}_0 < \bar{y}_1 < \bar{y}_2 < \ldots < \bar{y}_{N-K-1}$.
- (2) Partitioning: $Y \cap \bar{Y} = \phi$ *and* $Y \cup \bar{Y} = \{0, 1, 2, \ldots, N-1\}$.

Define two sets X and $\bar{X}$ such that $X = \{0, 1, 2, \ldots, K-2, K-1\}$ and $\bar{X} = \{K, K+1, K+2, \ldots, N-2, N-1\}$. Then, for the Flip Network, $r(Y, X)$ and $r(\bar{Y}, \bar{X})$ are satisfied by $\{r(y_i, i)\}_{i=0}^{K-1}$ and $\{r(\bar{y}_i, N-i-1)\}_{i=0}^{N-K-1}$.

Proof See [8]. ∎

Corollary 3.2 Lemma 3.1 holds with X and $\bar{X}$ defined as $X = \{0 \oplus v, 1 \oplus v, 2 \oplus v, \ldots, K-2 \oplus v, K-1 \oplus v\}$ and $\bar{X} = \{K \oplus v, K+1 \oplus v, K+2 \oplus v, \ldots, N-2 \oplus v, N-1 \oplus v\}$ where v is any integer such that $0 \leq v \leq N-1$ and $\oplus$ is addition modulo N. $r(Y, X)$ and $r(\bar{Y}, \bar{X})$ are satisfied by $\{r(y_i, i \oplus v)\}_{i=0}^{K-1}$ and $\{r(\bar{y}_i, (N-1-i) \oplus v)\}_{i=0}^{N-K-1}$.

Proof See [8]. ∎

Lemma 3.1 and Corollary 3.2 state that the Flip Network is nonblocking in the multi-channel sense for $(Y, \bar{Y}; X, \bar{X})$ wherein Y and $\bar{Y}$ are each monotone and X and $\bar{X}$ are each compact modulo N. The relationship between y_i and $\bar{y}_j$ is arbitrary for $i = 0, 1, 2, \ldots, K-1$ and $j = 0, 1, 2, \ldots, N-K-1$.

Figure 1 shows an example of the dispositions in $r(Y, X)$ and $r(\bar{Y}, \bar{X})$ according to Lemma 3.1 (or equivalently Corollary 3.2 with $v = 0$). In the figure, $Y = \{0, 1, 3, 6\}$, $\bar{Y} = \{2, 4, 5, 7\}$, $X = \{0, 1, 2, 3\}$, and $\bar{X} = \{4, 5, 6, 7\}$, which result in the specific connections of $r(0,0), r(1,1), r(3,2), r(6,3)$ for $r(Y, X)$ and $r(2,7), r(4,6), r(5,5), r(7,4)$ for $r(\bar{Y}, \bar{X})$. At the input, each input channel is given a cell, represented by the channel number at which it appears and the destined channel group X or $\bar{X}$. At the output, the cells are shown according to the connection patterns established in the switch.

One significant consequence of Lemma 3.1 and Corollary 3.2 is that the connection patterns are specified for both $r(\bar{Y}, \bar{X})$ and $r(Y, X)$. To observe this in detail, consider for example, Figure 2 where two possible connection patterns are shown in a Flip Network for which the dispositions of $r(\bar{Y}, \bar{X})$ are different. (Solid lines inside switching elements are required for $r(Y, X)$; dotted lines correspond to connection patterns in support of $r(\bar{Y}, \bar{X})$.) The existence of more than one disposition of the connection patterns which support $r(\bar{Y}, \bar{X})$ as illustrated in Figure 2 indicates that this set of connections is somewhat arbitrary. Consequently, suppose that cells belonging to a session are routed in the Flip Network as part of $r(\bar{Y}, \bar{X})$ without the connection patterns fully specified, then, after passing through the network, they appear at the output in some unspecified order as exemplified in Figure 2. Thus the relative ordering among the cells at $\bar{Y}$ cannot be easily preserved at $\bar{X}$.

On the other hand, consider again, in Figure 1, the cells appearing as part of $r(Y, X)$ at input channels $(y_0, 0), (y_1, 0), (y_2, 0), \ldots, (y_{K-1}, 0)$ which connect to $(x_0, n), (x_1, n), (x_2, n), \ldots, (x_{K-1}, n)$ according to Lemma 3.1. Hence $x_i = i$. The relative ordering among the cells at the input is preserved at the output since $y_0 < y_1 < y_2 < \ldots < y_{K-1}$ resulted in $x_0 < x_1 < x_2 < \ldots < x_{K-1}$. On the other hand, cells appearing at input channels $(\bar{y}_0, 0), (\bar{y}_1, 0), (\bar{y}_2, 0), \ldots, (\bar{y}_{N-K-1}, 0)$ as part of $r(\bar{Y}, \bar{X})$ connect to $(\bar{x}_0, n), (\bar{x}_1, n), (\bar{x}_2, n), \ldots, (\bar{x}_{N-K-1}, n)$ with $\bar{x}_i = N - i - 1$. The relative ordering at the output is now reversed since $\bar{y}_0 < \bar{y}_1 < \bar{y}_2 < \ldots < \bar{y}_{N-K-1}$ resulted in $\bar{x}_0 > \bar{x}_1 > \bar{x}_2 > \ldots > \bar{x}_{N-K-1}$.

At a time slot, since cells belonging to the same session can arrive simultaneously at many input channels, it is imperative to keep track of the relative ordering among the cells; if this were not possible, the time ordering among the cells would not be known unless each cell was provided with a time-stamp at the traffic source. We show in Section 4 how the relative ordering resulted by Lemma 3.1, *reversed or not*, can be used within the overall scheme of multi-channel switching to guarantee the in-order cell sequencing.

In order to apply the binary grouping defined in Lemma 3.1, it is necessary to assign each cell an element in X or $\bar{X}$ according to the position in which it appears at the inputs. For each cell belonging to $r(Y, X)$, starting from the top an element of X will be given in an increasing order. (The first one from the top gets 0, the next one gets 1, and so on.) For each cell belonging to $r(\bar{Y}, \bar{X})$, starting from the top an element of $\bar{X}$ will be given in a decreasing order. (The first one from the top gets $N-1$, the next one gets $N-2$, and so on.)

Assigning an element of X or $\bar{X}$ is accomplished by inserting the Running Adder (RA) before the Flip Network as shown in Figure 3. We term the combination of the RA and the Flip Network as the Nonblocking Binary Group Network (NBGN) whose name depicts its main function, viz., separating inputs into two groups in a non-blocking fashion. In Figure 4, an implementation of the RA for $N = 8$ is shown.

4 Nonblocking Multi-Channel Switching

To illustrate how more than two channel groups can be separated based on the construct of Lemma 3.1, we consider, as an example, the network shown in Figure 5 where four channel groups $00, 01, 10$, and 11 are defined. The corresponding four group connection requests are represented by $r(Y_{00}, X_{00}), r(Y_{01}, X_{01}), r(Y_{10}, X_{10})$, and $r(Y_{11}, X_{11})$.

To determine the specific dispositions in the first NBGN, cells arriving at the inputs are grouped based on the first bit (i.e., the rightmost bit) of the channel group address (e.g., 0 for X_{00} and X_{10}, and 1 for X_{01} and X_{11}). This implies that in the first NBGN, the four original group connection requests are aggregated to $r(Y_{00} \cup Y_{10}, X_{00} \cup X_{10})$ and $r(Y_{01} \cup Y_{11}, X_{01} \cup X_{11})$. In Figure 5, $X = X_{00} \cup X_{10} = \{0, 1, 2, \ldots, 7\}$ and $\bar{X} = X_{01} \cup X_{11} = \{8, 9, 10, \ldots, 15\}$. Then cells are grouped at the output of the first NBGN as shown in Figure 5.

The separation into four channel groups is completed by the second NBGN. This time, however, cells are grouped based on the second bit of the channel group address, resulting in two aggregated group connection requests, $r(Y_{00} \cup Y_{01}, X_{00} \cup X_{01})$ and $r(Y_{11} \cup Y_{10}, X_{11} \cup X_{10})$.

For the set of binary numbers whose bits are used in binary grouping, it is interesting to observe the sequence by which they appear after cascaded use of NBGN's. Following the present example, at each NBGN, groups with 0 at the corresponding address bit are considered as part of $r(Y, X)$ and groups with 1 address bit are part of $r(\bar{Y}, \bar{X})$. After the first NBGN, groups using binary number 0 appear with the higher relative ordering than groups using binary number 1, hence 0-1 sequence. After the second NBGN, the result is groups with 00 first, 01 second, 11 third, and finally groups with 10. The resulting sequence is 00-01-11-10. We show below that these sequences are a specific form of Gray code [2].

Let $\mathcal{G}$ represent the d-bit Gray code constructed as follows: Write down the binary numbers from 0 to $2^d - 1$ in an ascending order. Corresponding to each binary number, $b = b_{d-1}b_{d-2}...b_0$, in this list is the codeword, $g = g_{d-1}g_{d-2}...g_0$, of $\mathcal{G}$ where $g_{d-1} = b_{d-1}$ and $g_h = b_h \otimes b_{h+1}$, for $0 \le h \le d-2$. ($\otimes$ implies addition modulo 2.)

Lemma 4.1 Let g^i, $i = 0, 1, 2, \ldots t-1$, be the d-bit number assigned to channel group i for the purpose of binary grouping. Let $o^0, o^1, o^2, \ldots, o^{t-1}$ be an ordering of $g^0, g^1, g^2, \ldots, g^{t-1}$ according to occurrence in $\mathcal{G}$, i.e., $o^0, o^1, o^2, \ldots, o^{t-1}$ is a subsequence of G. Clearly, $o^i = g^{\pi(j)}$ for some permutation, π, of $\{0, 1, 2, \ldots, t-1\}$. Then at the end of d NBGN's, channel group $\pi^{-1}(0)$ has the highest relative ordering, channel group $\pi^{-1}(1)$ next, channel group $\pi^{-1}(2)$ next, ..., and finally, channel group $\pi^{-1}(t-1)$ at the bottom.

Proof The lemma can be proved by induction on t. It is easy to see that the assertion is true for t $=2$. Now assume it to be true for $t = s$. We need to show that it must be true for $t = s+1$. Now let k, $1 \le k \le t$, be such that every o^i in $O^0 = \{o^0, o^1, o^2, \ldots, o^k\}$ has its leftmost bit, o^i_{d-1}, equal to 0 and every o^i in $O^1 = \{o^{k+1}, \ldots, o^{t-1}\}$ has its leftmost bit equal to 1. Let the $(s+1)$th channel group be assigned to it the number g^{s+1}. Let $o^l = g^{s+1}$, i.e., $\pi(s+1) = l$.

Consider the case where $o^l \in O^0$. Since $o^0, o^1, o^2, \ldots, o^l, \ldots, o^k$ is a subsequence of $\mathcal{G}$, there must be p $(0 \le p \le d-1)$ so that o^i for any $i < l$, has $o^i_j = o^l_j = 0$ for $j = p, p+1, \ldots, d-1$ and $o^i_{p-1} = 0$ and $o^l_{p-1} = 1$. Therefore, regardless of their positions at the output of the $(p-1)^{th}$ NBGN (where o^i_{p-2} is used for grouping), the p^{th} NBGN (where o^i_{p-1} is used instead) routes cells such that the cells that are given o^i will have higher relative ordering than those given o^l at the output for $i < l$. Since $o^i_j = o^l_j = 0$ for $j = p, p+1, \ldots, d-1$, any subsequent application of NBGN does not change the relative ordering among the cells that are given any of $\{o^i\}_{i=0}^{l}$. Thus $o^0, o^1, o^2, \ldots, o^{l-1}$ have the higher relative ordering than o^l at the output of the d^{th} NBGN. By considering o^j for $j > l$, we can also see that all cells with o^j will have lower relative ordering than those with o^l. Thus, the lemma is seen to be true for $t = s + 1$. If we assume $o^l \in O^1$, a similar argument can be asserted for $t = s + 1$. ∎

Consider an N-input network and suppose it is desired to group outputs into t channel groups. Input channels which do not have a cell and consequently are not actually destined to any output are assumed to be assigned to a fictitious channel group which is included in t. At a specific instance of time, let $n_0, n_1, n_2, \ldots, n_{t-1}$ be the number of input channels with cells destined for channel groups $0, 1, 2, \ldots, t-1$ respectively where $\sum_{i=0}^{t-1} n_i = N$.

Let $\lceil log_2 t \rceil = d$. Separation of the t channel groups can be satisfied by applying the method described in Lemma 3.1 repeatedly, d times. Let $b^0 < b^1 < b^2 < \ldots < b^{t-1}$ be d-bit binary numbers where $b^i = b^i_{d-1} b^i_{d-2} \ldots b^i_0$, is the address for channel group i. Define Gray code $g^i = g^i_{d-1} g^i_{d-2} \cdots g^i_0$ such that $g^i_{d-1} = b^i_{d-1}$ and $g^i_h = b^i_h \otimes b^i_{h+1}$, for $0 \leq h \leq d-2$. Then Lemma 4.1 states that if g^i is used for channel group i in routing through d cascaded NBGN's, the sequence in which cells belonging to different channel groups arrive at the output is such that output channels $0, 1, \ldots, n_0 - 1$ receive cells belonging to channel group 0, output channels $n_0, n_0+1, \ldots, n_0+n_1-1$ receive cells belonging to channel group 1, …, outputs $\sum_{j=0}^{i-1} n_j, \sum_{j=0}^{i-1} n_j + 1, \ldots, \sum_{j=0}^{i-1} n_j + n_i - 1$ receive cells belonging to channel group i, …, and output channels $\sum_{j=0}^{t-2} n_j, \sum_{j=0}^{t-2} n_j + 1, \ldots, \sum_{j=0}^{t-2} n_j + n_{t-1} - 1$ receive cells belonging to channel group $t-1$. The sequence of channel group addresses appearing at the output is $b^0 - b^1 - b^2 - b^3 - \ldots - b^{t-1}$ which use the sequence of Gray codewords $g^0 - g^1 - g^2 - g^3 - \ldots - g^{t-1}$ for the purpose of binary grouping.

Each time cells go through an NBGN, binary grouping is done based on a single bit in their Gray codewords and for those cells with this bit equal to 1, the relative ordering among them is reversed. In order to maintain the in-order cell sequencing, we find it convenient to utilize g^i's such that each g^i has an even number of 1's; in this way, cells belonging to the same session have the same relative ordering at the final output. The relative ordering at the final output for some channel groups will be reversed if the Gray codes given to them have an odd number of 1's. For example, Figure 5 shows that the relative orderings for channel groups that used 01 and 10 in binary grouping are reversed at the output while the same relative orderings are observed for channel groups that used 00 and 11.

Lemma 4.2 Let $\tilde{g}$ be a Gray codeword constructed from $\tilde{b} = \tilde{b}_{\tilde{d}-1} \tilde{b}_{\tilde{d}-2} \ldots \tilde{b}_0$ such that $\tilde{g}_{d-1} = \tilde{b}_{\tilde{d}-1}$ and $\tilde{g}_h = \tilde{b}_h \otimes \tilde{b}_{h+1}$, for $0 \leq h \leq \tilde{d} - 2$. Then, $\tilde{g}$ has an even number of 1's if and only if $\tilde{b}_0 = 0$, i.e., $\tilde{b}$ is an even number.

Proof Write down the binary numbers from 0 to $2^{\tilde{d}} - 1$ in ascending order. Starting from 000…00 in this list, generate the sequence of Gray codewords according to the lemma. First we note that binary number 000…00 generates Gray codeword 000…00 which has an even number of 1's. Since two consecutive entries in the sequence of Gray codewords differ in exactly one position, the number of 1's in any two Gray codewords separated by one entry between them is both either even or odd. Since binary number 000…00 has an even number of 1's, every even binary number has an even number of 1's in the Gray codeword it generates. ∎

For the t channel groups being considered, $t-1$ Gray codewords with even number of 1's are required. (We do not care about the relative ordering among the empty cells.) There are $2^{\tilde{d}-1}$ even numbers among $\tilde{d}$-bit binary numbers. To support t channel groups, $2^{\tilde{d}-1} \geq t-1$, which results in $\tilde{d} = \lceil log_2 t - 1 \rceil + 1$. The fictitious group is assigned 111…11 as the binary channel group address, which is an odd number. (The fictitious group is separated to the bottom.) Thus at most one additional NBGN is required to separate t groups while maintaining the relative ordering within each group.

Figure 6 shows an example of a Gray codeword sequence in which each of the first eight codewords has an even number of 1's. The last codeword has an odd number of 1's to be used for the empty inputs.

The interconnection pattern between two consecutive NBGN's in the cascade allows recursive use of the same NBGN with a single cell buffer separating two recursions. We term the combination of the NBGN and the buffer as the Recursive Nonblocking Binary Group Network (RNBGN). An RNBGN is shown in Figure 7.

As a result, an internally nonblocking multi-channel switch is realized by the Flip Network of $log_2 N$ stages; Figure 7 is a multi-channel switch that is internally nonblocking in the multi-channel sense and that supports an arbitrary number ($\leq N$) of channel groups (including the single-channel mode) by simply controlling the number of recursions performed over the single NBGN. An N-input switch with 2^r channel groups requires $r + 1$ recursive uses of the NBGN to maintain the relative ordering.

5 The Architecture

We now present the overall architecture of our multi-channel switch. An efficient way to provide the in-order cell sequencing is to equate at each time slot, the time ordering among the cells belonging to the same session with their relative ordering and maintain this relative ordering at each switching entity throughout the network. Then, within a channel group, a lower indexed link (one that appears above) has the higher time ordering than do higher indexed links at each time slot. An illustration of this is shown in Figure 8 using a sequence of cells generated by a single session at $t_0 < t_1 < t_2 < \ldots$.

The overall architecture of the switch is best viewed as a two phase structure involving the Multicasting Phase followed by the Routing Phase. When a cell arrives at the input, VPI/VCI is extracted from its header which determines the required connection in the switch. The first stage of connection is the Multicasting Phase which generates an appropriate number of copies of the cell. The main function of the Routing Phase is to deliver these copies to the correct destinations.

The details of the Multicasting Phase are shown in Figure 9. A cell arriving at the input channel i is processed by the Port Processor (PP) i; PP i extracts the VPI/VCI from the cell and determines from a table stored in it the required number of copies. The cell is augmented by three additional fields, namely *Broadcast Connection ID*, *Cardinality*, and *Sum*. Then PP i records in the *Broadcast Connection ID* field a unique identifier given to this VPI/VCI during the connection set up time; the recorded value in the *Broadcast Connection ID* field is maintained throughout the switch and is used to determine the required information for all functions of switching. In the *Cardinality* field, the number of copies to be made for this *Broadcast Connection ID* is recorded. PP i also records in the *Sum* field 0 for all active cells. Empty cells are assigned 1 in the *Sum* field.

The augmented cell now enters the $(N + R_M)$-input NBGN (R_M is the number of input channels reserved for recirculation which is described shortly). Based on the value stored in the *Sum* field, the NBGN perform the binary grouping, and thus, active cells are concentrated at the upper portion of the output with the relative ordering maintained among them.

After passing through the NBGN, the active cells enter the Address Encoder (AE) over a compact interval (i.e., all active inputs appear consecutively). The use of the AE is introduced in [4]. Following the illustrations in [4], the active cells are processed by the AE which determines whether or not the fan-out request of the cells can be accommodated by the Copy Network (CN). The AE calculates the running sum of the fan-outs starting from the top input; each cell is marked 0 in its *Sum* field if the running sum of fan-outs including its own does not exceed N, and otherwise, the cell is marked 1. From the running sum, the AE can also determine for each cell marked 0 the interval of the CN output to which the duplicated cells are routed. (This interval must also be attached to the cell, which is not shown explicitly here.)

Since the fan-out requests for the cells belonging to the same session are identical, the AE may skip over a cell violating the fan-out limit in order to allow cells below it to be passed through. It can be seen that this maintains the time ordering among the cells; if the fan-out request of a cell is rejected, all fan-out requests of subsequent cells belonging to that session would be rejected during the same time slot.

Cells are then routed to the second NBGN where according to the *Sum* field, they are either passed to the CN (if the *Sum* value is 0) or routed to the recirculation paths (if the *Sum* value is 1) being fed back to the R_M inputs of the NBGN reserved for recirculation. The recirculated cells are tried again at the next time slot at which time their priorities have been increased due to the positions (i.e., they appear at the top) at the input of the NBGN; they are given higher priorities over the cells entering the switch for the first time since the running sum at the AE always starts from the top. Maintaining the relative ordering among the recirculated cells is accomplished by twisting recirculation paths before connecting back to the input. (This is due to the fact that the relative ordering among the recirculated cells are reversed by the second NBGN.)

The cells passed to the CN are duplicated and appear at the output interval determined by the AE. The CN is identical to the nonblocking copy network suggested by [4] where duplicated copies of a cell appear consecutively at the output.

In the Routing Phase, the throughput of a switch is constrained by contentions among inputs vying for the limited number of output channels in a channel group, even if the switch is internally nonblocking. Of the buffering schemes discussed in the literature, it is known that the shared buffering scheme achieves the optimal delay-throughput performance with a small number of buffers.

Consider the switching architecture in Figure 10 where R_R of the $(N + R_R)$ inputs available in the RNBGN are reserved for recirculation. Cells targeted for various channel groups enter at the input and iterate through the RNBGN. After completing the required number of recursions, they appear at the output of the RNBGN according to a Gray coded sequence of channel group addresses.

Cells enter the NBGN where the contentions for output are identified and excess cells are marked for recirculation. The NBGN separates the excess cells from the rest while maintaining the required relative ordering. (The Running Adder function of this NBGN is different from the rest but we omit the detailed discussion.) The cells passed to the Banyan Router (BR) can be routed to the proper destinations without further possibility of contention since the BR is nonblocking with respect to compact inputs that are destined for *monotone ascending* output addresses. On the other hand, the excess cells are recirculated for attempts at the next time slot at which time, they are given priority due to the higher relative ordering of the recirculation paths. Such a mechanism ensures that the in-order cell sequencing is maintained.

6 Numerical Results

To verify the validity of the switching paradigm proposed in Section 5, we conducted a series of simulations. The Multicasting Phase shown in Figure 9 is simulated for $N = 64$ with the fan-out of cells given as independent, identically distributed, truncated geometric random variables. In the simulations, the AE is allowed to skip cells that exceed the running sum of N in order to permit the cells below to be copied. Results of the simulations for throughput, cell loss probability, and delay are shown in Figure 11 as function of the normalized offered load, ρ. ρ is defined to be the average fan-out of cells times the probability that there is a cell at each input channel.

In the figures, average fan-out of 2 was assumed. We observe that for $R_M \geq 64$, cell lost probability can be kept less than 10^{-6} for ρ equals 0.9. Even for $R_M = 8$, cell lost probability of 10^{-6} can be realized for $\rho \approx 0.6$. The delay performance shown in Figure 11 corresponds to the average number of times that a cell is recirculated before it is copied. We observe that in the region where the cell loss probability is less than 10^{-6}, cells are subject to insignificant levels of delay.

We repeated the simulations for average fan-out = 32. Results are shown in Figure 12 where the performance levels of throughput, cell loss probability, and delay are worse. Nevertheless, cell loss probability of 10^{-6} can still be achieved for $\rho \approx 0.85$ if there is a sufficient number of recirculation paths (say $R_M \geq 64$). The delay performance of cells are shown to be reasonable (say less than 5) at such operating conditions.

We also conducted a simulation study for the Routing Phase of the switch in Figure 10. Once again, $N = 64$ is assumed. Input channels have independent, identically distributed cells entering the Routing Phase of the switch and destinations of these cells are equally likely among all channel groups. At each time slot, there is a cell arriving at any input channel with probability equals to the input utilization.

Results of the simulations for throughput, cell loss probability, and delay are shown in Figure 13 as function of the input utilization (i.e., the probability that there is a cell at any time slot for input). The results shown in Figure 13 correspond to the worse case scenarios where there is no performance gain due to channel grouping (i.e., the number of channel groups = N). We see that for $R_R = 64$, it is possible to support the input utilization level of about 0.65 for cell loss probability of 10^{-6}. For $R_R = 128$, the input utilization increases to greater than 0.75 and for $R_R = 256$, it becomes greater than 0.85. Figure 13 illustrates that cells do not experience excessive delay in the region where cell loss probability is acceptable.

Even if we introduce a small level of channel grouping, performance gain is notable in the Routing Phase. Figure 14 present the results of simulations with the number of channel groups = 32. We see that for cell loss probability of 10^{-6}, the input utilization becomes 0.74, 0.85, and 0.92 for $R_R = 64, 128$ and 256, respectively. Equally notable improvements are reported for the delay performance.

7 Conclusions

When there is significant difference between transmission speed over optical fibers and the processing speed of electronic switches, multi-channel switching can play a significant role in alleviating the bottleneck at the switching node. We have presented an efficient architecture for multi-channel switching that enables multicasting while at the same time guaranteeing in-order cell sequencing.

References

[1] Cruz, R.L., "The Statistical Data Fork: A Class of Broadband Multichannel Switches," *IEEE Transactions on Communications*, vol. 40, no. 10, pp. 1625-1634, October 1992.

[2] Kohavi, Z., *Switching and Finite Automata Theory*, McGraw-Hill, 1978.

[3] Lau, P.S. and Leon-Garcia, A., "Design and Analysis of a Multilink Access Subsystems Based on the Batcher-Banyan Network Architecture," *IEEE Trans. Commun.*, vol. 40, No. 11, Nov. 1992, pp. 1757-1766.

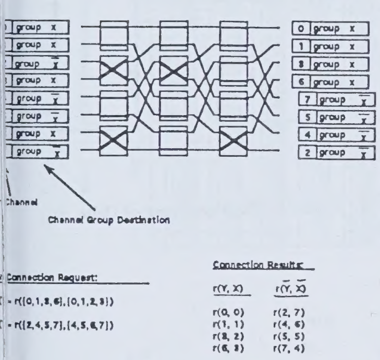
[4] Lee, T. T., "Nonblocking copy Networks for Multicast Packet Switching," *IEEE J. Select. Areas Commun.*, vol. 6, No. 9, Dec. 1988, pp. 1455-1467.

[5] Leighton, F.T., *Introduction to Parallel Algorithms and Architectures: Arrays, Trees, Hypercubes*, Morgan Kaufmann Publishers, 1992.

[6] Lin, A., and Silvester, J.A., "Fixed-Node Routing and Its Performance in ATM Networks," *Proceedings of IEEE INFOCOM '90*, June, 1990, pp. 803-810.

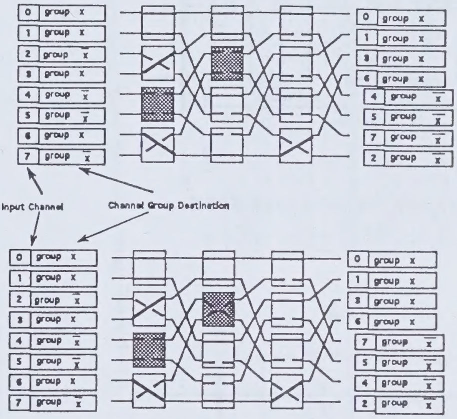
[7] Pattavina, A., "Multichannel Bandwidth Allocation in a Broadband Packet Switch," *IEEE Journal of Selected Areas in Communications*, Volume 6, No. 9, Dec, 1988, pp. 1489-1499.

[8] Saidi, H., Min, P.S. and Hegde, M.V., "A Nonblocking Architecture for Broadband Multi-Channel Switching," *Technical Report*, Department of Electrical Engineering, Washington University, 1993.



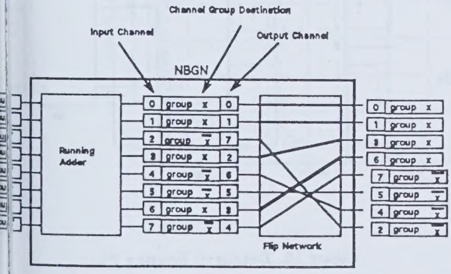
group x
group x̄
0 group x
1 group x
3 group x
6 group x
7 group x̄
5 group x̄
4 group x̄
2 group x̄
Channel
Channel Group Destination
Connection Request:
= r({0,1,3,6},{0,1,2,3})
= r({2,4,5,7},{4,5,6,7})
Connection Results:
r(Y, X)
r(Ȳ, X̄)
r(0, 0)
r(1, 1)
r(3, 2)
r(6, 3)
r(2, 7)
r(4, 6)
r(5, 5)
r(7, 4)

gure 1. An Example of $r(Y,X)$ and $r(\bar{Y},\bar{X})$



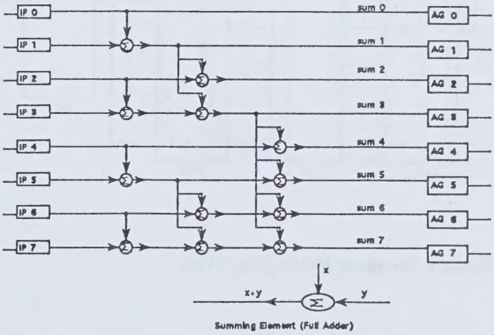
0 group x
1 group x
2 group x̄
3 group x
4 group x̄
5 group x̄
6 group x
7 group x̄
0 group x
1 group x
3 group x
6 group x
4 group x̄
5 group x̄
7 group x̄
2 group x̄
Input Channel
Channel Group Destination
0 group x
1 group x
3 group x
6 group x
7 group x̄
5 group x̄
4 group x̄
2 group x̄

Figure 2. Two Instances of Grouping



Channel Group Destination
Input Channel
Output Channel
NBGN
Running Adder
0 group x 0
1 group x 1
2 group x̄ 7
3 group x 2
4 group x̄ 6
5 group x̄ 5
6 group x 3
7 group x̄ 4
Flip Network
0 group x
1 group x
3 group x
6 group x
7 group x̄
5 group x̄
4 group x̄
2 group x̄

Figure 3. Nonblocking Binary Group Network



IP 0
IP 1
IP 2
IP 3
IP 4
IP 5
IP 6
IP 7
sum 0
sum 1
sum 2
sum 3
sum 4
sum 5
sum 6
sum 7
AG 0
AG 1
AG 2
AG 3
AG 4
AG 5
AG 6
AG 7
x
x+y
y
Σ
Summing Element (Full Adder)

Figure 4. Running Adder

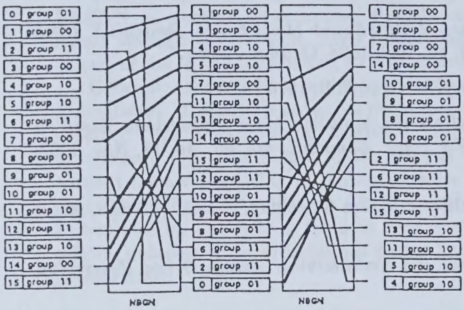
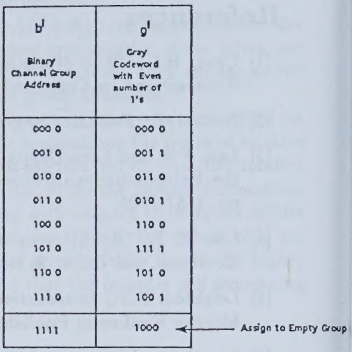

0 group 01
1 group 00
2 group 11
3 group 00
4 group 10
5 group 10
6 group 11
7 group 00
8 group 01
9 group 01
10 group 01
11 group 10
12 group 11
13 group 10
14 group 00
15 group 11
1 group 00
3 group 00
4 group 10
5 group 10
7 group 00
11 group 10
13 group 10
14 group 00
15 group 11
12 group 11
10 group 01
9 group 01
8 group 01
6 group 11
2 group 11
0 group 01
1 group 00
3 group 00
7 group 00
14 group 00
10 group 01
9 group 01
8 group 01
0 group 01
2 group 11
6 group 11
12 group 11
15 group 11
13 group 10
11 group 10
5 group 10
4 group 10
NBGN
NBGN

Figure 5. Separating Four Channel Groups



b^j Binary Channel Group Address	g^j Gray Codeword with Even number of 1's
000 0	000 0
001 0	001 1
010 0	011 0
011 0	010 1
100 0	110 0
101 0	111 1
110 0	101 0
111 0	100 1
1111	1000

Assign to Empty Group

Figure 6. Gray Code with Even Number of 1's

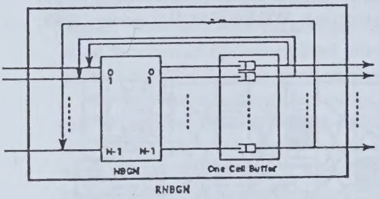

0
1
0
1
N-1
N-1
NBGN
One Cell Buffer
RNBGN

Figure 7. Recursive Use of Single NBGN

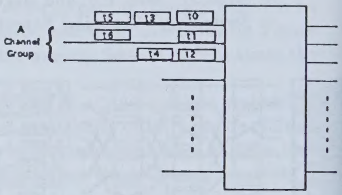

15
13
10
16
11
14
12
A Channel Group

Figure 8. Time Ordering of Cells

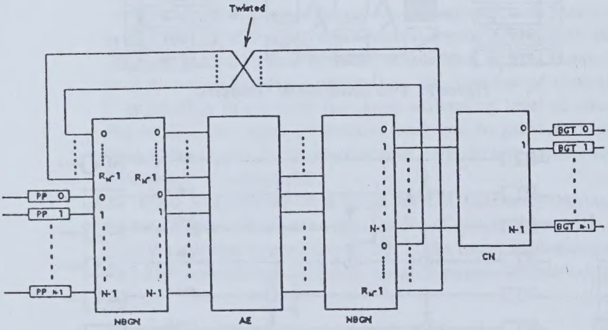

Twisted
PP 0
PP 1
PP N-1
0
R_M-1
1
N-1
NBGN
AE
NBGN
CN
BGT 0
BGT 1
BGT N-1

Figure 9. Details of Multicasting Phase

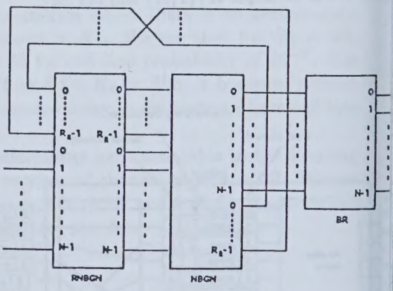

0
R_B-1
1
N-1
RNBGN
NBGN
BR

Figure 10. Details of Routing Phase

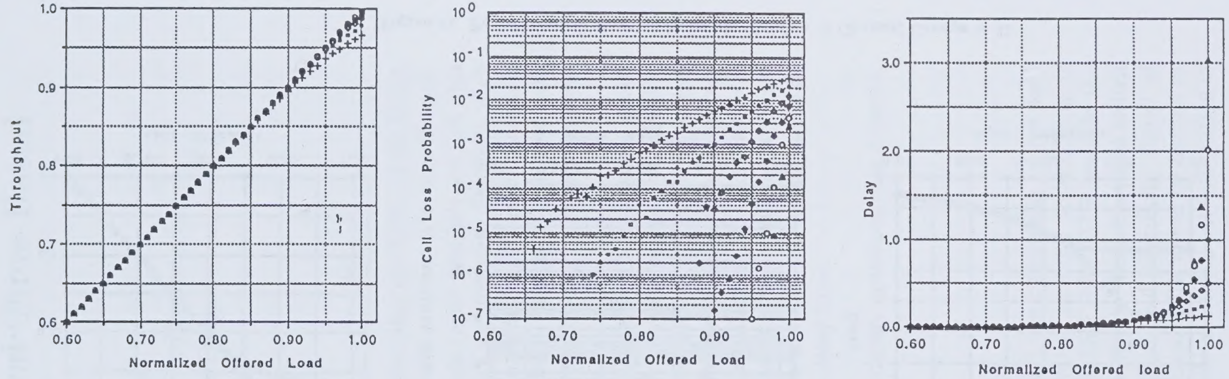


Figure 11. Performance of Copy Network: Average Fan-out = 2

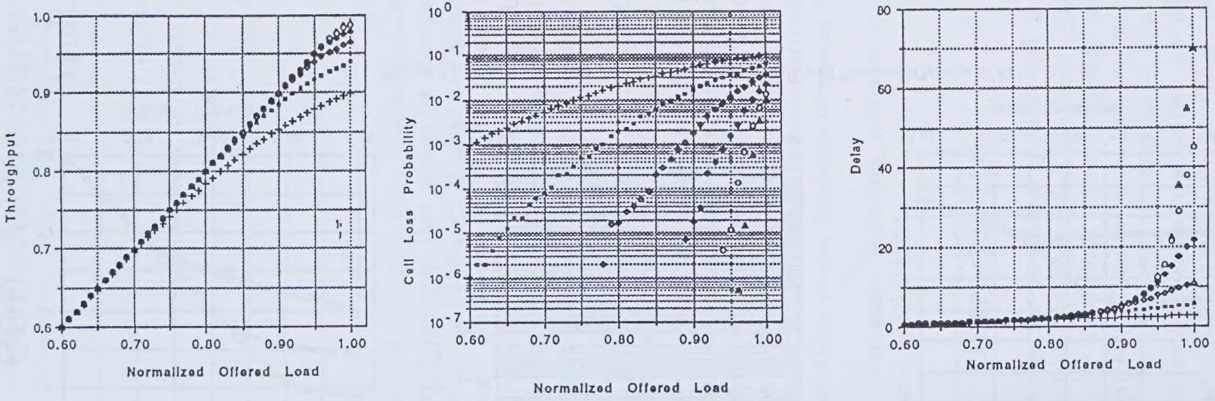


Figure 12. Performance of Copy Network: Average Fan-out = 32

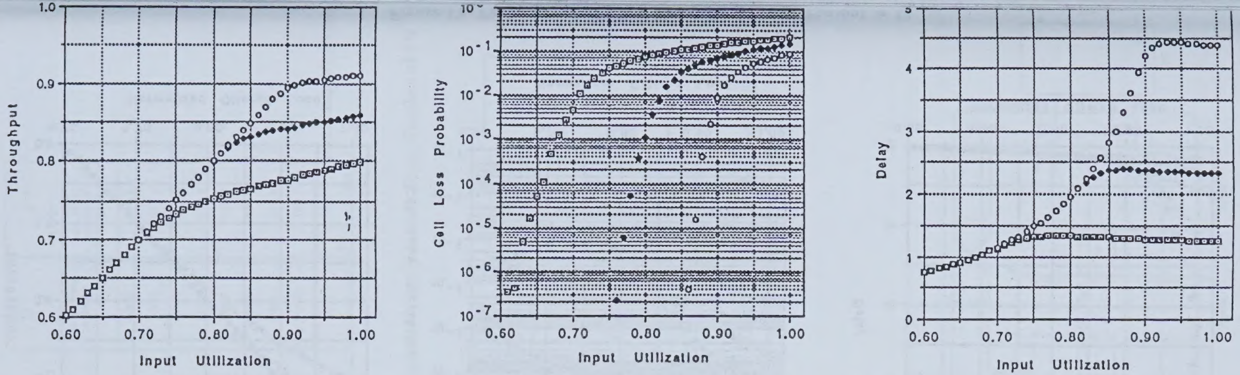


Figure 13. Performance of Routing Network: Number of Channel Groups = 64

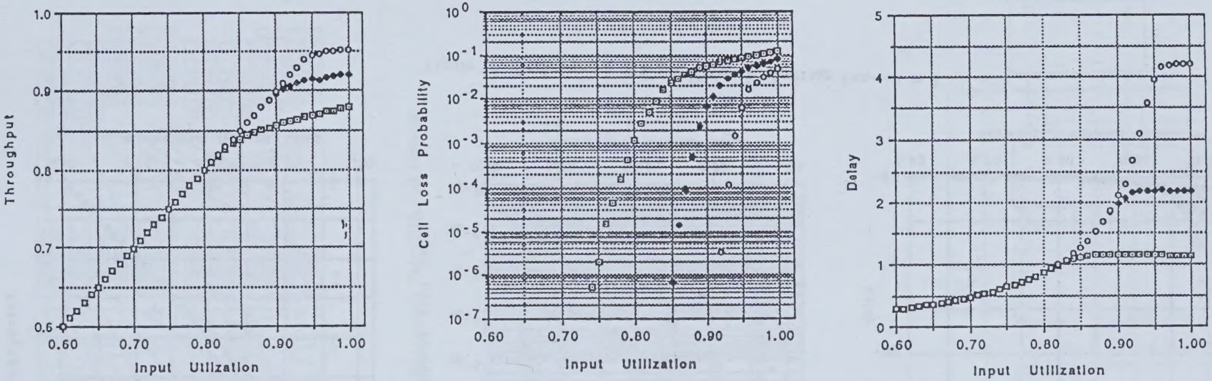


Figure 14. Performance of Routing Network: Number of Channel Groups = 32

COD: Architectures for Optical Time-Based Multiplexers *

R. L. Cruz J. T. Tsai
Department of Electrical & Computer Engineering
University of California, San Diego
La Jolla, CA 92093-0407

Abstract

Current architectures for time-based multiplexers are approaching the limits of electronic switching speed, thus raising the question of how best to utilize recent advances in optoelectronic technology in order to efficiently construct high speed multiplexers. In this paper, we introduce *Cascaded Optical Delay line* (COD) architectures for ultra high speed multiplexing and concentration. The COD architectures utilize an extremely simple distributed electronic control algorithm; Preliminary feasibility studies indicate that with technology available off the shelf today, a COD multiplexer can operate at speeds in excess of 10^9 *packets* per second.

A fundamental issue in the design of multiplexers is dealing with contention, which arises when two or more packets arrive to the multiplexer at the same time. Packet contention is resolved by using optical 2×2 switches, whose states are driven by optical power detectors, to switch packets to optical fiber delay lines which provide a temporary memory function. For example, with only eleven 2×2 switches, a 2-to-1 multiplexer can be constructed with a packet loss probability less than 10^{-15} at a utilization factor of 80% under memoryless packet arrivals. Bursty packet arrivals can be supported at essentially no cost by simply increasing the length of the delay lines.

Keywords: Statistical multiplexing, high speed switching, optoelectronic systems, bursty traffic, ATM.

1 Introduction

Due to the abundance of bandwidth in optical fibers and the complexity of (packet) switching at high speed, a natural approach towards optical network architectures is to use a broadcast oriented topology, and move the switching complexity towards the periphery of the network. However, it is not clear how well this "broadcast and switch at the periphery" approach scales with the number of users and geographic extent of the network. Aside from the fact that the total network bandwidth would be limited to that of a single optical fiber, a difficult problem becomes the management/calibration of wavelengths (or spreading codes, in the case of CDMA). There are two problems here, first in the allocation and synchronization of wavelengths to avoid transmitter contention, and second in how the receivers know what wavelength to tune to.

---

*This work was supported by the National Science Foundation under grants NCR 89-04029, NCR 91-58618, and NCR 93-18416, with matching donations from MIL3 Inc. and Clearpoint Research Corp.

It thus appears probable that large scale optical networks (e.g. hundreds of millions of users) will involve switching in time as well as in space (wavelength space or code space). Thus, although we believe that large scale optical networks will involve WDM and/or CDMA, we also believe that they will involve some sort of time based switching.

In this paper, we focus on a fundamental problem in time based switching, namely multiplexing. For our purposes, time based multiplexers are systems which take several asynchronous optical packet streams at the input, and merge these input streams in time onto a single, higher rate, packet stream. Each input stream occupies a nominal band of wavelengths, and no wavelength conversion occurs from input to output. The multiplexers have the constraint that only one packet may exit at a time. A major difficulty in the design of multiplexers is dealing with contention. Contention arises between different packet streams if two or more packets arrive at the multiplexer at the same time. In this paper, we present *Cascaded Optical Delay lines* (COD) architectures, which utilize novel configurations of optical delay lines to provide a temporary buffering mechanism for resolving packet contention.

An application for an optical time based multiplexer would be in implementing the output queue of output queued packet switches [11]. Another possible application for a time-based multiplexer would be to resolve contentions at the front end of a user's wavelength tunable receiver in a WDM network so that at most one packet is received in each time slot [3].

2 Model and Basic Building Blocks

In this paper we will consider only a simplified model. We assume that all packets are of equal length and have a constant transmission time. For example, the packet transmission time might be one nanosecond. To encode the bits of a packet so that they can be transmitted in a time as short as one nanosecond, one could use ultra-short optical pulses and delay lines, as discussed in [10]. Alternatively, the bits of a packet can be encoded in separate wavelengths [2]. The delay line, 2×2 switch, and the packet detector are the basic building blocks we will consider to construct optical multiplexers.

The delay line is simply a segment of optical fiber. It functions as an optical memory element. Obviously, this type of memory element is quite different than its electronic counterpart, because of the fixed storage time determined by the speed of light. Using optical time domain reflectometers, the length of delay lines can be controlled quite precisely. For example, assuming the speed of propagation is 2×10^{10} centimeters per second, a packet whose transmission time is 1 nanosecond requires a delay line of 20cm for one packet time of storage. A packet may have to be delayed (stored) if contention arises. Packets can be switched to delay lines by using 2×2 switches. Some packet switching architectures utilizing optical delay lines can be found in [7] [9] [4] [5] [3] [6],[10].

The 2×2 switch has two inputs and two outputs. Light entering each input can exit on exactly one of the two outputs according to the state of the switch, which can be either in the "bar" state or in the "cross" state. One of the inputs (outputs) is labeled top, the other bottom. While in the bar state, the top (bottom) input is connected to the top (bottom) output; while in the cross state, the top (bottom) input is connected to the bottom (top) output. The state of the switch is controlled by an electronic control

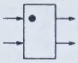


Figure 1: A Lane Changer (LC) Module. Packets "desire" bottom output (lane).

signal, and can change at rates in excess of one billion times per second with commercially available switches based on lithium niobate technology.

To generate the electronic signals required for driving the 2 × 2 switches, we need to determine whether packets are present at the inputs of the switches. We can detect the presence of packets by splitting off a small fraction of the signal power via a directional coupler and feeding it into a high speed photodetector. The optical power detector generates an electrical signal which indicates the presence of optical power and hence the presence of a packet. The photodetectors' speed is capacitively limited; detectors are readily available that can respond to frequencies in excess of 10 GHz. Packet detectors constructed in this way will work for any type of modulation technique used for encoding bits of information in the packet, except perhaps for polarization modulation.

In this paper, we will consider only the case of synchronous operation. That is, we assume that time is divided into slots of equal duration; the length of a slot is the transmission time of a single packet (e.g. 1 nanosecond). We assume that packet transmissions begin only on slot boundaries. Furthermore, we consider here only the ideal situation in which packet detection is instantaneous, the switching time required for a 2 × 2 switch to change state is zero, and no propagation delays except for those introduced by delay lines.

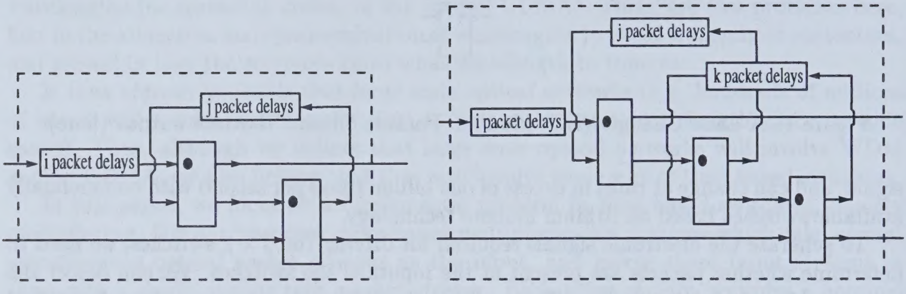
2.1 Recursive Lane Changer Modules

In this section, we define the *Lane Changer* (LC) module as a basic building block. *Recursive Lane Changer* (RLC) modules will then be defined in terms of LC modules. In the next section, we present the COD architectures in terms of these building blocks.

The LC module is an optical 2 × 2 switch together with a packet (optical power) detector, and is illustrated in Figure 1. If power (i.e. a packet) is detected at the top input marked with a dot, then the switch is set to the cross state; otherwise it is set to the bar state. One may in fact think of the LC module as a 2-to-1 multiplexer in itself with no buffering; The bottom output is the output of this 2-to-1 multiplexer, and lost packets exit on the top output. If packets enter the LC module on both inputs in the same slot, then the top input is given priority to reach the bottom output.

In the following it is instructive to make an analogy with cars. One may interpret Figure 1 in the context of two "lanes" of a highway, the "top lane" and the "bottom lane." The bottom lane is desired for all cars. Cars which enter on the top lane gravitate to the bottom lane, and the LC module facilitates this (hence the name "Lane Changer"). Cars entering on the bottom lane will stay on the bottom lane, unless they get "pushed up" (deflected) to the undesirable top lane by the presence of a car which enters on the top lane.

The LC module is not a very good multiplexer in itself; Since it has no buffering, packets will be "lost" rather frequently. The Recursive Lane Changer module of type (i, j) is defined schematically in Figure 2(a). It has two delay lines and two LC modules. The first delay line holds i packets, and the second delay line holds j packets. The RLC



(b) JT-RLC (i,j,k)

Figure 2: (a) "Recursive Lane Changer" of Type (i, j). (b) "Joined Twin - Recursive Lane Changer" of Type (i, j, k).

module is in fact a type of Lane Changer module in itself, since cars in the top lane will gravitate towards the bottom lane, possibly at the expense of cars in the bottom lane. It can easily be seen that for any packet passing through an RLC module of type (i, j), the delay suffered is at most $\max\{i, j\}$, and that the packet passes through at most three 2×2 switches.

The *Joined Twin - Recursive Lane Changer* (JT-RLC) Module of type (i, j, k) is defined schematically in Figure 2(b). As indicated in the figure, the JT-RLC module of type (i, j, k) has 3 delay lines which hold i,j,and k packets each. It can easily be seen that for any packet passing through a JT-RLC module of type (i, j, k), the delay suffered is at most $\max\{i, j + k\}$, and that the packet passes through at most five 2×2 switches.

3 COD Architectures

In this section we present multiplexer architectures defined in terms of the building blocks of the previous section. We begin by considering 2-to-1 multiplexers.

3.1 A Baseline 2-to-1 Multiplexer

We define a baseline 2-to-1 multiplexer with n stages to be the cascade connection of an LC module followed by n RLC modules of type $(0, 1)$, where the top lanes are connected to top lanes and the bottom lanes are connected to bottom lanes. A baseline 2-to-1 multiplexer with $n = 3$ stages is illustrated in Figure 3. It can be verified that a baseline 2-to-1 multiplexer with n stages behaves as a *first in - first out* "work conserving" multiplexer with a storage capacity of n packets, where the oldest packet is discarded upon buffer overflow. (To verify this, it is helpful to note that at each RLC module, a packet entering on the top lane implies that another packet must enter the the same RLC module on the bottom lane in the same slot.)

The queueing analysis of the baseline 2-to-1 multiplexer is thus standard. In particular, if packets arrivals occur independently in each slot on each input link with probability

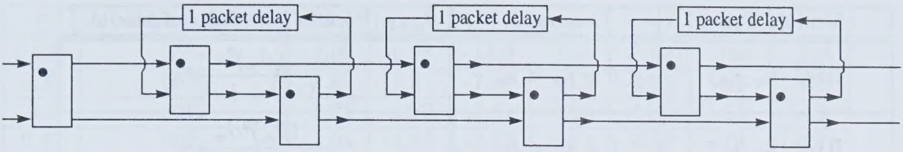


Figure 3: A baseline 2-to-1 multiplexer with $n = 3$ stages. Functionally equivalent to a FIFO work conserving multiplexer with a buffer capacity of n packets.

p, then the steady state packet loss probability is $L(n,p)$, defined as

$$L(n,p) = \frac{\beta(\frac{\beta}{1-\alpha})^n(1-\frac{\beta}{1-\alpha})}{(\alpha+\beta)\left[1-(\frac{\beta}{1-\alpha})^{n+1}\right]}, \tag{1}$$

where $\alpha = 2p - p^2$ and $\beta = p^2$.

It appears possible to monolithically integrate all of the 2 × 2 switches to reduce coupling loss. However, it is apparently quite difficult to obtain a totally monolithic implementation, due to the required lengths of delay lines. If it becomes possible, but still costly, the baseline multiplexer is attractive, since the required total length of delay lines is the smallest possible.

On the other hand, the 2 × 2 switches are at present of relatively high dollar cost. Perhaps more importantly in the long term, each 2 × 2 switch introduces a power loss of about 2 or 3 dB in the optical signals which pass through it. It can easily be seen that for the baseline architecture with n stages, a packet that is not lost and suffers a delay of k slots passes through $1+n+2k$ 2 × 2 switches. For moderate utilization factors and bursty input traffic, the number of required stages n is large, as well as the probable values of delay. This will cause unacceptable power losses unless optical amplifiers are used. However, optical amplifiers are presently of high cost and introduce additional noise, so it is desirable to avoid them when possible.

In summary, the baseline architecture has good functional properties (first in - first out, optimal delay), but requires many 2 × 2 switches and has high power loss. In the next subsection, we present some important refinements to the baseline architecture which result in *far* fewer required 2 × 2 switches, and smaller power loss, for the same performance (packet loss probability). These desirable qualities are obtained at the expense of packet fairness (first in - first out) and delay. However, from an engineering standpoint, the delay can still be made negligible compared to propagation delay in a wide area network. Furthermore, if a user's bandwidth is sufficiently small compared to the line rate of the multiplexer, then with high probability the user will "not notice" that the multiplexer is not FIFO; Out of sequence packets can be dropped at the destination with only a negligible increase in the packet loss rate.

3.2 Refined COD Architectures for 2-to-1 Multiplexers

Remarkably small packet loss probabilities can be obtained by appropriately changing the lengths of the delay lines in the baseline 2-to-1 multiplexer and/or using JT-RLC modules instead of RLC modules. We explore these refinements in this subsection.

Module Type	Comments	β_{m+1} (Pseudo Loss Rate of Module)
RLC (i_m, j_m)	$0 = i_m < j_m$	$\dfrac{(\beta_m)^2}{1-\alpha_m+\beta_m}$
RLC (i_m, j_m)	$0 < i_m < j_m$	$\dfrac{(\beta_m)^2\alpha_m}{1-\alpha_m(1-\beta_m)}$
JT-RLC (i_m, j_m, k_m)	$0 < i_m < j_m = k_m$	$\dfrac{\alpha_m(\beta_m)^3}{(1-\alpha_m)^2+\alpha_m\beta_m(1-\alpha_m+\beta_m)}$

Table 1: Pseudo Loss Rate β_{m+1} for various module types.

Consider a single RLC module of type (i_m, j_m), or a single JT-RLC module of type (i_m, j_m, k_m) , which is the m^{th} module in a cascade arrangement. Let $A_{m,k}$ and $B_{m,k}$ be indicator random variables (0-1 valued) which indicate packet arrivals on the bottom and top inputs to the RLC or JT-RLC module in the k^{th} slot, respectively. Similarly, let $A_{m+1,k}$ and $B_{m+1,k}$ be indicator random variables which indicate packet departures on the bottom and top outputs of the RLC or JT-RLC module in the k^{th} slot, respectively. Let $\alpha_m = Prob\{A_{m,k} = 1\}$ and $\beta_m = Prob\{B_{m,k} = 1\}$. In words, α_m and β_m are simply the average rates (packets/slot) at which packets enter the bottom and top inputs of the module. We will assume that an arrival on the top input implies an arrival on the bottom input in the same slot. In other words $Prob\{B_{m,k} = 1\} = Prob\{A_{m,k} = 1 \text{ and } B_{m,k} = 1\} = \beta_m$ and hence $Prob\{A_{m,k} = 1 \text{ and } B_{m,k} = 0\} = \alpha_m - \beta_m$.

Similarly, let α_{m+1} and β_{m+1} be the average rates that packets depart on the bottom and top outputs of the module, respectively, i.e. $\alpha_{m+1} = Prob\{A_{m+1,k} = 1\}$ and $\beta_{m+1} = Prob\{B_{m+1,k} = 1\}$. Note that a departure on the top output implies a departure on the bottom output in the same slot; Thus $Prob\{B_{m+1,k} = 1\} = Prob\{A_{m+1,k} = 1 \text{ and } B_{m+1,k} = 1\} = \beta_{m+1}$ and hence $Prob\{A_{m+1,k} = 1 \text{ and } B_{m+1,k} = 0\} = \alpha_{m+1} - \beta_{m+1}$.

We now discuss the "input-output" behavior of this RLC or JT-RLC module. Specifically we are interested in expressing $(\alpha_{m+1}, \beta_{m+1})$ as a function of (α_m, β_m). We call β_{m+1} the pseudo loss rate of the module; It will be equal to the loss rate of the multiplexer if the module is the last one in the chain. Clearly, we would like β_{m+1} to be small.

Suppose packet *departures* from the *delay lines* in the module in any given slot k are statistically independent of $A_{m,k}$ and $B_{m,k}$. It turns out that if we make this independence approximation, we can in fact express $(\alpha_{m+1}, \beta_{m+1})$ as a function of (α_m, β_m). It can be shown that with this independence assumption, we obtain the values indicated for β_{m+1} in Table 1. Note that α_{m+1} is also determined from this table, since by conservation of flow we have $\alpha_{m+1} = \alpha_m + \beta_m - \beta_{m+1}$.

The independence approximation is in fact accurate provided the delay lines in the module are long with respect to the "time constant" of the arrival process to the module. Indeed, departures from the delay lines in the module in a given slot then depend on the arrivals to the delay lines in much earlier slots, which in turn are almost independent of arrivals to the module in the given slot.

The appearance of the factor $(\beta_m)^2$ or $(\beta_m)^3$ in Table 1 is significant. It in fact implies that, provided the independence approximation is accurate, the *logarithm* of the loss probability grows *exponentially* in the number of modules in the chain which make up the multiplexer. To illustrate this, Table 2 gives the base 10 logarithm of the packet loss probability for a chain consisting of an LC module followed by n RLC modules, where it is assumed that $i_m = 0$ in the m^{th} RLC module for all m. Packet arrivals on each

p	Utilization Factor	$n = 1$	$n = 2$	$n = 3$	$n = 4$	$n = 5$	$n = 6$	$n = 7$
0.125	25%	-2.90	-6.28	-13.0	-26.6	-53.6	-108	-216
0.25	50%	-1.90	-3.81	-7.63	-15.3	-30.5	-61	-122
0.4	80%	-1.21	-1.99	-3.41	-6.23	-11.9	-23	-45
0.45	90%	-1.04	-1.55	-2.33	-3.74	-6.53	-12	-23

Table 2: log_{10} of packet loss probability for a 2-to-1 multiplexer consisting of an LC module followed by n RLC modules with $0 = i_m < j_m$ in each module (independence approximation).

p	Utilization Factor	$n = 1$	$n = 2$	$n = 3$	$n = 4$	$n = 5$	$n = 6$	$n = 7$
0.125	25%	-3.52	-8.13	-17.3	-35.8	-72.6	-146	-293
0.25	50%	-2.23	-4.78	-9.86	-20.0	-40.4	-81.0	-162
0.4	80%	-1.35	-2.34	-4.19	-7.87	-15.2	-30	-59
0.45	90%	-1.14	-1.75	-2.71	-4.54	-8.18	-15	-30

Table 3: log_{10} of packet loss probability for a 2-to-1 multiplexer consisting of an LC module followed by n RLC modules with $0 < i_m < j_m$ in each module (independence approximation).

input of the first LC module are assumed memoryless with arrival rate p. The values in this table were simply obtained by using the top row of Table 1 recursively, with the initial condition $\beta_1 = \beta = p^2$ and $\alpha_1 = \alpha = 2p - p^2$. The packet loss probability is then $Prob\{\text{packet loss}\} = \beta_{n+1}/(2p)$. It is interesting to note in this case that

$$Prob\{\text{packet loss}\} = L(2^n - 1, p), \tag{2}$$

where L was defined in (1) (this can be proven by induction on n). Thus, assuming the independence approximation is accurate, this refined architecture with n RLC modules has the same loss probability as the baseline 2-to-1 multiplexer with $2^n - 1$ modules.

Tables 3 and 4 are similar to Table 2; In Table 3 it is assumed that $i_m > 0$ in each of the n RLC modules. In Table 4 it is assumed that each module is a JT-RLC module where $0 < i_m < j_m = k_m$ for each of the n JT-RLC modules. Tables 3 and 4 were obtained by recursively using the second and third rows of Table 1, respectively.

Comparing Tables 2 and 3, we see the benefit of having the first delay line of non-zero length ($i_m > 0$) in an RLC module. The first delay line of each module avoids the sure contention that would otherwise arise when a packet arrives on the top input of the module.

Comparing Tables 3 and 4, we see the benefit of using JT-RLC modules instead of

p	Utilization Factor	$n = 1$	$n = 2$	$n = 3$	$n = 4$	$n = 5$
0.125	25%	-5.22	-17.2	-53.1	-161	-484
0.25	50%	-3.19	-9.88	-29.9	-90.1	-270
0.4	80%	-1.75	-4.22	-11.6	-33.6	-99.7
0.45	90%	-1.40	-2.74	-6.37	-17.3	-49.9

Table 4: log_{10} of packet loss probability for a 2-to-1 multiplexer consisting of an LC module followed by n JT-RLC modules with $0 < i_m < j_m = k_m$ in each module (independence approximation).

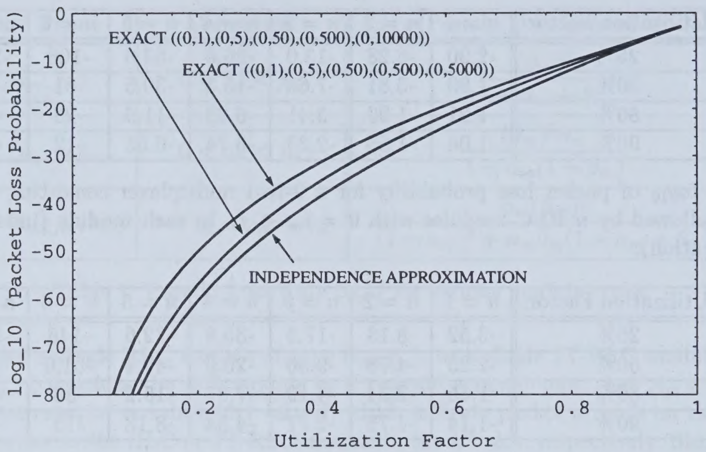


Figure 4: $\log_{10}$ of packet loss probability versus utilization factor.

RLC modules. To make a fair comparison, we should take into account that each RLC module uses only 2 LC modules, whereas each JT-RLC module uses 3 LC modules. Thus, for example, the $n = 6$ column of Table 3 can be compared with the $n = 4$ column of Table 4.

We now briefly explore how the independence approximation can be made accurate by proper choice of the lengths of the delay lines. Given a multiplexer constructed by the cascade of an LC module followed by n RLC modules, where the m^{th} RLC module is of type (i_m, j_m), we say the *delay line profile* is $((i_1, j_1), (i_2, j_2), \ldots, (i_n, j_n))$. Figure 4 plots the base 10 logarithm of the *exact* packet loss probability versus the utilization factor $\rho = 2p$, for $n = 5$ and the delay line profiles of $((0,1),(0,5),(0,50),(0,500),(0,5000))$ and $((0,1),(0,5),(0,50),(0,500),(0,10000))$. These exact values of the packet loss probabilities were computed by solving a system of $5000 \cdot 2^n = 160,000$ equations and $10000 \cdot 2^n = 320,000$ equations, respectively- details of the analysis method are omitted here. Also plotted in Figure 4 are the corresponding results using the independence approximation, i.e. $L(2^n - 1, \rho/2)$ versus ρ. As can be seen from Figure 4, the independence approximation yields reasonably accurate results in both cases. Increasing the length of the delay line in the fifth stage from 5,000 to 10,000 improves the accuracy of the independence assumption; However, if one is interested in the maximum utilization at which the loss probability is less than, say 10^{-10}, then this increase in delay line length does not have much impact - in either case this maximum utilization is about 80%.

The numerical example above suggests that the delay lines must grow rather quickly to yield results consistent with the independence approximation. However, there are heuristics for choosing the delay line profile which may yield smaller delay lines for the same performance. One such heuristic is to choose the delay line lengths in different modules to be *relatively prime* numbers. Another heuristic is to arrange for the delay lines in modules $1, 2, \ldots, m-1$ to be multiples of an integer which is relatively prime with respect to the delay line lengths in module m, for each m. Unfortunately, although simulation results indicate that these heuristics yield excellent performance with only

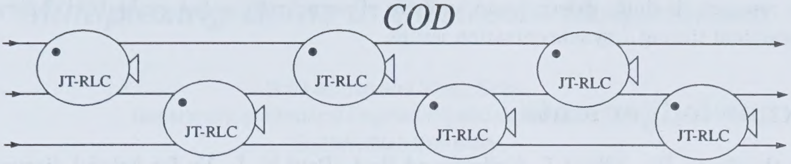


Figure 5: A 3-to-1 multiplexer. Also functions as a 3-to-2 concentrator.

moderately long delay lines, the only exact analysis method we have in this case is a brute force solution of the underlying Markov chain, which is prohibitive when the total length of the delay lines is larger than about 15. (The exact analysis mentioned in the preceding paragraph relied heavily on the delay line lengths being harmonically related).

3.3 Architectures for N-to-1 Multiplexers and Concentrators

We can utilize the same ideas to construct multiplexers with more inputs. Of course, we can construct such multiplexers by making a "tree" of 2-to-1 multiplexers. However, there are other potentially better constructions which exploit statistical sharing of buffers to a greater degree. As an example, suppose we wish to construct a 3-to-1 multiplexer. We can visualize 3 "lanes" instead of 2 lanes as we had done above. By inserting RLC or JT-RLC modules between lanes at various points, we can make traffic flow towards the bottom lane. An example is shown in Figure 5. For an N-to-1 multiplexer where N is large, it probably makes sense to put lane changer modules between non-adjacent lanes, perhaps in a "banyan" type of interconnection pattern. We leave this for future investigation. It is interesting to note that the architecture in Figure 5 also functions as a 3-to-2 concentrator, where packets from the three input links are "merged" with high probability onto the bottom two lanes. Such concentrators are useful for constructing multi-channel switches [1] [8].

3.4 Bursty Traffic

We have also considered bursty input traffic. Specifically, we considered the case where each input packet stream is a two-state Markovian on-off source, where the holding time in each state is geometrically distributed, packets are always generated in the on state, and no packets are generated in the off state. Evidently, essentially the same performance (loss probability) can be achieved as for the memoryless case by simply increasing the length of each delay line. It seems that making sure each delay line can hold at least a few mean burst lengths of packets is sufficient. This is a rather inexpensive way of dealing with bursty traffic, as compared to conventional multiplexing approaches, since the cost (in dollars) of increasing the length of optical fiber delay lines is negligible. This idea has been reported before by Haas in [5]

4 Closing Remarks

We have presented some architectures for high speed time-based multiplexers and concentrators, which have a variety of applications. There are many possible directions for future research, including design and analysis of asynchronous (i.e. unslotted) operation and practical timing / synchronization issues.

Acknowledgements

Many thanks to Dr. Albert L. Kellner and Prof. Paul K. L. Yu for helpful discussions which have significantly influenced this work.

References

[1] B. Bellur and G. Sasaki, "Buffered Statistical Data Fork," *Elect. Engg. Res. Lab.*, The Univ. of Texas at Austin, Tech. Rep. C-90-2, Oct. 1990.

[2] D. J. Blumenthal, K. Y. Chen, J. Ma, R. J. Feuerstein, and J. R. Sauer, "A Deflection Routing 2x2 Photonic Switch for Computer Interconnects," preprint, Oct. 1991, submitted to IEEE Photonics Technology Letters.

[3] I. Chlamtac and A. Fumagalli, "QUADRO-Star: High Performance Optical WDM Star Networks," *Proc. IEEE GLOBECOM'91*, December 1991.

[4] Z. Haas, "The 'Staggering Switch': An Almost-all Optical Packet Switch," ," *Proc. IEEE GLOBECOM'92*, December 1992.

[5] Z. Haas, "Extensions to the 'Staggering Switch' Architecture," *Proc. IEEE INFOCOM'93*, pp. 455–463, March 1993.

[6] F. Masetti, P. Gavignet-Morin, D. Chiaroni, and G. Da Loura, "Fiber Delay Lines Optical Buffer for ATM Photonic Switching Applications," *Proc. IEEE INFOCOM '93,* pp. 935–942, March 1993.

[7] P. R. Prucnal, M. A. Santoro, and S. K. Sehgal, "Ultrafast All-Optical Synchronous Multiple Access Fiber Networks," *IEEE JSAC*, vol. SAC-4, no. 9, pp. 1484-1493, Dec. 1986.

[8] H. Saidi, P. S. Min, and M. V. Hegde, "A Nonblocking Architecture for Broadband Multi-Channel Switching," Department of Elect. Eng., Washington University, St. Louis, Technical Report WUEE-93-135 / WUCCRC-93-09, March 15, 1993.

[9] S. Ramanan, H. Jordan, and J. Sauer, "A New Time Domain, Multistage Permutation Algorithm," *IEEE Trans. Information Theory,* vol. 36 no. 1, pp. 171–173, Jan. 1990.

[10] Y. Shimazu and M. Tsukada, "Ultrafast Photonic ATM Switch with Optical Output Buffers," *Journal of Lightwave Technology,* vol. 10 no. 2, pp. 265-272, Feb. 1992.

[11] Y. S. Yeh, M. Hluchyj, and S. Morgan, "The Knockout Switch: A Simple Architecture for High-Performance Packet Switching," *IEEE J. Select. Areas Commun.*, vol. 5 no.8, pp. 1274–1283, Oct. 1987.

Multiplexing Gains in Bitstream Multiplexors

Scott Jordan and Ikhlaq Sidhu
Department of Electrical Engineering and Computer Science
Northwestern University
Evanston, IL 60208-3118
scott@eecs.nwu.edu, ikhlaq@eecs.nwu.edu

ABSTRACT

We examine the capability of a fixed capacity high speed bit pipe to multiplex a moderate number of bitstreams under a low loss constraint. The analysis is based on a more general model of multiple service, multiple resource loss systems, with the service type defined by the instantaneous distribution of bit rate. We consider three schemes of allocating capacity between services and show that more services can be accommodated with schemes which allow more multiplexing. We are in particular interested in the variation of different multiplexing gains with changes in burstiness.

I. Introduction

Based on the Multiple Source, Multiple Resource (MSMR) model[Jord91] [JordPr], we consider a channel of capacity C as a resource which can accommodate certain combinations of service types (A, B, C... etc.) subject to a resource allocation constraint. The constraint is set by a maximum probability of overload, PB_{max}. An acceptance region is the set of all combinations of numbers of various service types which satisfy this constraint. We might think of the types A,B,C as different services provided by the network such as video, voice, data respectively and that different numbers of each service type can satisfy the constraint on blocking probability.

In the development of "effective bandwidth" [Hui88] [Kell91], the acceptance region for the unbuffered and buffered cases is shown to be linearly bounded under various performance guarantees. Each service type has an associated effective bandwidth such that the sum of all effective bandwidths for all multiplexed sources must be less than the capacity of the channel. Many studies [ElwaMi] assign effective bandwidths to buffered models operating in the asymptotic regime of small loss probability. Our model does not include a buffer. This model is suited for the condition when the source bit rates are very high and a comparatively short buffer will have little effect.

In this paper, we show a simple method to approximate the resource allocation constraint for the two service type model of multiplexing. We analyze the approximated constraint to gain insight into the nature and controlling parameters of the true resource allocation constraint. Although, the constraints can be given for the multiple service type model, we can more easily gain insight by examining the two source model.

II. Bit Stream Multiplexing Model

In this section we define a bit stream multiplexing model which includes three levels of multiplexing/sharing. A schematic of the model is given in Figure 1.

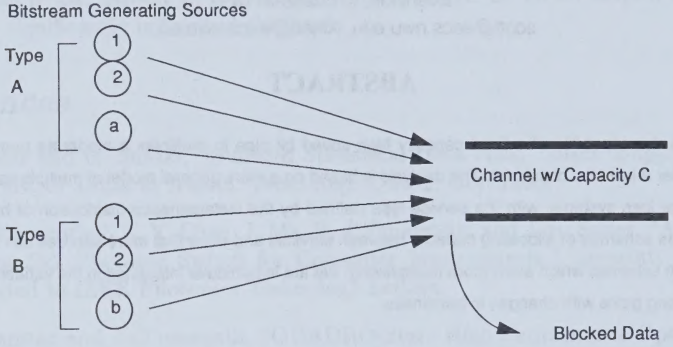


Figure 1. Bit Stream Multiplexing Block Diagram

Bit stream sources generate data (in bits or packets) at a time-varying rate. Sources are classified by a service type, A or B, which is defined by an instantaneous data rate distribution, $f_A(r)$ or $f_B(r)$ respectively. Sources of type A and B are enumerated (1, 2,... a) and (1, 2,... b) respectively. We assume that each source has a data rate which is an ergodic random process and that the sources are independent of each other.

Define: S = {A,B}, set of service types; a = number of type A, b = number of type B; $a \geq 0, b \geq 0$

Define: N = [a, b], operating point, vector of number of each source type.

Define: $R_{A,n}(t) \sim f_A(r)\ \forall n \in \{1 \ldots a\}$, data rate process of the n^th A source.

Define: $R_{B,n}(t) \sim f_B(r)\ \forall n \in \{1 \ldots b\}$, data rate process of the n^th B source.

In accordance with an allocation discipline, data streams from all sources share a channel with fixed capacity C. We define blocking to be the event that some portion of the combined data stream is not accepted by the channel.

We now present three alternative disciplines to allocate the channel between sources. These disciplines are presented in decreasing order of resource sharing / multiplexing: 1) complete sharing between all sources, 2) allocation by source type and 3) allocation by individual service.

In complete sharing, all source types share an entire channel. Blocking occurs only when the combined data stream is greater than the channel capacity C as detailed in EQ 1.

$$C < \sum_{n=1}^{a} R_{A,n}(t) + \sum_{n=1}^{b} R_{B,n}(t) \qquad \text{(EQ 1)}$$

In allocation by source type, the channel of capacity C is partitioned into C_A and C_B and each sub-channel is devoted for its respective service type. For example an A type source can not use capacity C_B even if C_A is fully utilized and C_B is free. Blocking occurs when either or both partitions are overloaded (EQ 2).

$$C_A < \sum_{n=1}^{a} R_{A,n}(t) \qquad or \qquad C_B < \sum_{n=1}^{b} R_{B,n}(t) \qquad \text{(EQ 2)}$$

In allocation by individual service, we further partition the channel such that each subchannel is dedicated to one particular source. There is no sharing of capacity between sources in the multiplexing discipline. Type A services are allocated a capacity of $\tilde{C}_A$ and type B services are allocated a capacity of $\tilde{C}_B$. Blocking occurs if any source exceeds its allocated capacity (EQ 3).

$$\tilde{C}_A < R_{A,n}(t) \text{ for any } n \in \{1 \ldots a\} \qquad \text{or} \qquad \tilde{C}_B < R_{B,n}(t) \text{ for any } n \in \{1 \ldots b\} \qquad \text{(EQ 3)}$$

III. Resource Allocation Constraint for Complete Sharing

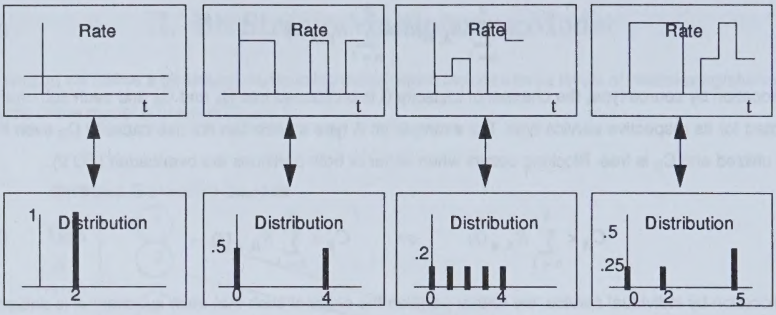
For a specific channel capacity C with complete sharing and maximum blocking probability PB_{max}, a resource allocation constraint can be found using $f_A(r)$ and $f_B(r)$. In this section, we illustrate this constraint for 3 sample cases. We build on these examples as the paper continues.

Since the combined data rate R(t) for some fixed t is the sum of independent random variables, the distribution $f_R(r)$ of R(t) can be expressed by the following convolution equation:

$$R(t) = \sum_{n=1}^{a} R_{A,n}(t) + \sum_{n=1}^{b} R_{B,n}(t) \qquad \text{(EQ 4)}$$

$$f_R(r) = f_A(r)^{(a)} * f_B(r)^{(b)} \qquad \text{(EQ 5)}$$

As an illustrative example, we select parameters C = 100 and $PB_{max} = 0.001$. Using the examples sources of Figure 2, the constraint was numerically evaluated for the following three cases of source type pairs: i) deterministic vs. on/off, ii) on/off vs. uniform, iii) uniform vs. arbitrary



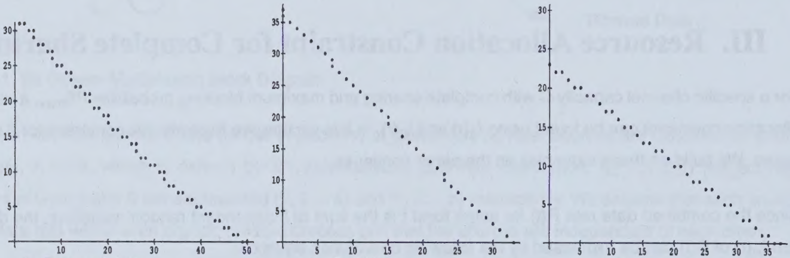
Deterministic:
mean = 2
variance = 0

On/Off 50%:
mean = 2
variance = 4

Uniform:
mean = 2
variance = 2

Arbitrary:
mean = 3
variance = 4.5

Figure 2: Time varying data rates and corresponding first order density functions.



A: Deterministic
B: On/Off

A: On/Off
B: Uniform

A: Uniform
B: Arbitrary

Figure 3. Constraint of maximum blocking probability on allocation of services for three example source pairs.

In the above graphs the x-axis gives the number of type A sources and the y-axis gives the number of type B sources which may be multiplexed together. Points above and/or right of the plotted points do not satisfy the constraint on blocking probability and thus do not provide the acceptable level of service. If the number of type A sources is fixed at zero, then the point on the y-axis shows how many type B sources can be accommodated by the channel with the prescribed quality of service. By reducing the number of type B sources, we see that more type A sources can be accommodated. We note that for the deterministic data rate (variance = zero), the number of allowable sources is the channel capacity divided by the (mean) data rate. A source type with the same mean but larger variance is more bursty. As we expect, fewer of the more bursty sources can be accommodated.

IV. Closed Form Approximation of Resource Allocation Constraint for Complete Sharing

Although the resource allocation constraint can be obtained using convolutions as shown in the last section, the method is calculation intensive, and furthermore, a closed form result is not available. In this section we obtain a closed form expression to approximate the exact resource allocation constraint. A preliminary discussion of the behavior and accuracy of the approximating expression is given. Also, the example constraints from the last section are overlaid with constraints obtained through the approximating closed form expression.

We specify the mean source rates, μ_A and μ_B, and the sources rate variances, σ_A^2 and σ_B^2, for type A and B services respectively. The mean and variance of the combined data stream rate R(t) in EQ 4 are μ_R and σ_R^2.

$$\mu_R = a \cdot \mu_A + b \cdot \mu_B \qquad \text{(EQ 6)}$$

$$\sigma_R^2 = a \cdot \sigma_A^2 + b \cdot \sigma_B^2 \qquad \text{(EQ 7)}$$

The distribution $f_R(r)$ is obtained through self convolutions of $f_A(r)$ and $f_B(r)$. By the central limit theorem, we know that $f_R(r)$ approaches a Gaussian distribution as a and b increase. By approximating $f_R(r)$ to be a Gaussian with mean and variance specified by EQ 6 and EQ 7, we are able to obtain an approximation of the constraint.

$$\Phi\left(\frac{C - \mu_R}{\sigma_R}\right) \leq 1 - PB_{max}$$

$$C \leq a \cdot \mu_A + b \cdot \mu_B + \rho\sqrt{a \cdot \sigma_A^2 + b \cdot \sigma_B^2} \qquad where \qquad \rho = \Phi^{-1}(1 - PB_{max}) \qquad \text{(EQ 8)}$$

Denote this continuous function (at equality) of (a,b) the *constraint curve*. Note that to specify a specific constraint curve, only the source rate means and variances are needed for a given channel capacity (C) and service quality (PB_{max} or ρ). The entire distribution for each source is not required.

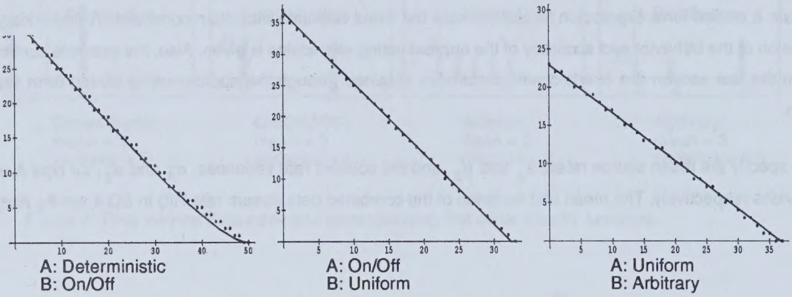
We separate the variables a and b in the constraint curve expression by holding one constant and solving a quadratic equation for the other. In Figure 4, the examples from the previous section have been overlaid with corresponding constraint curves using EQ 9.

$$\begin{aligned} a &= P_a - b \cdot Q_a - \frac{1}{2}\sqrt{R_a + b \cdot S_a} \\ b &= P_b - a \cdot Q_b - \frac{1}{2}\sqrt{R_b + a \cdot S_b} \end{aligned} \qquad \text{(EQ 9)}$$

where

$$P_a = \frac{C}{\mu_A} + \frac{\rho^2\sigma_A^2}{2\mu_A^2} \qquad Q_a = \frac{\mu_B}{\mu_A} \qquad R_a = \frac{4\rho^2 C\sigma_A^2}{\mu_A^3} + \frac{\rho^4\sigma_A^4}{\mu_A^4} \qquad S_a = \frac{4\rho^2}{\mu_A^2} \cdot \left(\sigma_B^2 - \sigma_A^2 \cdot \frac{\mu_B}{\mu_A}\right)$$

$$P_b = \frac{C}{\mu_B} + \frac{\rho^2\sigma_B^2}{2\mu_B^2} \qquad Q_b = \frac{\mu_A}{\mu_B} \qquad R_b = \frac{4\rho^2 C\sigma_B^2}{\mu_B^3} + \frac{\rho^4\sigma_B^4}{\mu_B^4} \qquad S_b = \frac{4\rho^2}{\mu_B^2} \cdot \left(\sigma_A^2 - \sigma_B^2 \cdot \frac{\mu_A}{\mu_B}\right)$$



A: Deterministic
B: On/Off

A: On/Off
B: Uniform

A: Uniform
B: Arbitrary

Figure 4. Exact resource allocation constraint overlaid with approximate constraint curves for three examples.

We show that the region outlined by the constraint curve has a convex complement in the first quadrant. As a result, the constraint curve will be a straight line or bend in towards the origin.

Theorem 1:

Approximating constraint curve has convex complement in first quadrant if $PB_{max} < 0.5$.

Proof:

Consider C in EQ4 as a function of a and b with probability PB_{max} is fixed and specified by ρ. C(a,b) is then a surface in 3-dimensional space with C(a,b) represented as height z-axis above the x,y plane. This surface is convex cap in the first quadrant since the Hessian of C(a,b) is negative definite for $\rho > 0$ (i.e $PB_{max} < 0.5$).

$$\begin{bmatrix} a & b \end{bmatrix} \begin{bmatrix} \frac{\partial^2}{\partial a^2} C(a,b) & \frac{\partial^2}{\partial a \partial b} C(a,b) \\ \frac{\partial^2}{\partial b \partial a} C(a,b) & \frac{\partial^2}{\partial b^2} C(a,b) \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} -a^2\sigma_A^4 - ab\sigma_A^2\sigma_B^2 - ba\sigma_A^2\sigma_B^2 - b^2\sigma_B^4 \end{bmatrix} \frac{\rho}{4} (a \cdot \sigma_A^2 + b \cdot \sigma_B^2)^{\frac{3}{2}} \leq 0$$

Define P(C)={(a,b)| C(a,b)>C, a>0, b>0, C>0}, first quadrant complement of Acceptance Region

Convex Cap $C(a,b) \leftrightarrow P(C)$ is Convex

In many practical cases, the bend towards the origin is small and the curve can be still further approximated by a straight line which has the same axis intercepts as the constraint curve. We define such a line to be the *liberal linear constraint approximation* illustrated in Figure 5 below. We call this constraint liberal because points on this line may be outside of the constraint curve. Although, the maximum distance between the liberal constraint and the constraint curve is small, we find that it does increase as the source types become less similar. When the two source types are identical, the constraint curve becomes a line.

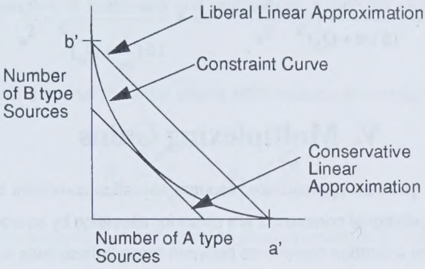


Figure 5. Liberal and conservative linear approximations of resource allocation constraint curve.

The axis intercept a' is easily obtained by setting b to zero in EQ 8. The resulting quadratic has two roots. The meaningful solution is given below. The axis intercept b' is obtained in a similar fashion. By inspecting the expressions for a' and b' we see that for sources with zero variance, a' = C/μ_A and b' = C/μ_B. As variance is increased, the intercepts decrease. By parameterizing mean, variance, service quality, and capacity into one parameter, K, we gain insight into this phenomenon.

$$\begin{aligned} a' &= \frac{C}{\mu_A} - \frac{1}{2}\left(\frac{\rho\sigma_A}{\mu_A}\right)^2\left(\sqrt{1+\frac{4C\mu_A}{\rho^2\sigma_A^2}} - 1\right) = \frac{C}{\mu_A}\cdot\left[1-\frac{1}{2}K_A\left(\sqrt{1+\frac{4}{K_A}}-1\right)\right] \\ b' &= \frac{C}{\mu_B} - \frac{1}{2}\left(\frac{\rho\sigma_B}{\mu_B}\right)^2\left(\sqrt{1+\frac{4C\mu_B}{\rho^2\sigma_B^2}} - 1\right) = \frac{C}{\mu_B}\cdot\left[1-\frac{1}{2}K_B\left(\sqrt{1+\frac{4}{K_B}}-1\right)\right] \end{aligned} \qquad \text{(EQ 10)}$$

where

$$K_A = \frac{\rho^2\sigma_A^2}{C\mu_A} \qquad K_B = \frac{\rho^2\sigma_B^2}{C\mu_B}$$

The slope of the liberal linear constraint is found by dividing the (negative) y-axis intercept by the x-axis intercept.

$$m = -\frac{\mu_A}{\mu_B}\cdot\frac{1-\frac{K_B}{2}\left(\sqrt{1+\frac{4}{K_B}}-1\right)}{1-\frac{K_A}{2}\left(\sqrt{1+\frac{4}{K_A}}-1\right)} \qquad \text{(EQ 11)}$$

We are also interested in the *conservative linear constraint* shown in Figure 5 because all points on that line satisfy the constraint curve. Since the area between the liberal linear constraint and constraint curve is convex, the curve is farthest away from the line at the point $(\hat{a}, \hat{b})$ where the slope of the curve and line are the same. The conservative linear constraint is a line with slope m which runs through $(\hat{a}, \hat{b})$, as shown in Figure 5.

$$m = \frac{db}{da} = -Q_b - \frac{S_b}{4\sqrt{R_b + \hat{a}S_b}} \qquad \frac{1}{m} = \frac{da}{db} = -Q_a - \frac{S_a}{4\sqrt{R_a + \hat{b}S_a}}$$

$$\hat{a} = \frac{S_b}{16\,(m+Q_b)^2} - \frac{R_b}{S_b} \qquad \hat{b} = \frac{S_a}{16\,(\frac{1}{m}+Q_a)^2} - \frac{R_a}{S_a}$$

V. Multiplexing Gains

In previous sections we have given an approximate resource allocation constraint for multiplexing with complete sharing. In this section, additional constraints are given for allocation by source type and individual service. We compare the resource allocation constraints between these 3 disciplines in multiplexing.

Recall the constraint for complete sharing:

$$C_{sharing} = a \cdot \mu_A + b \cdot \mu_B + \rho\sqrt{a \cdot \sigma_A^2 + b \cdot \sigma_B^2} \qquad \text{(EQ 12)}$$

We find a similar constraint for allocation by source type by using EQ 12 for C_A and C_B individually.

$$C_A = a \cdot \mu_A + \rho\sqrt{a \cdot \sigma_A^2} \qquad C_B = a \cdot \mu_B + \rho\sqrt{a \cdot \sigma_B^2}$$

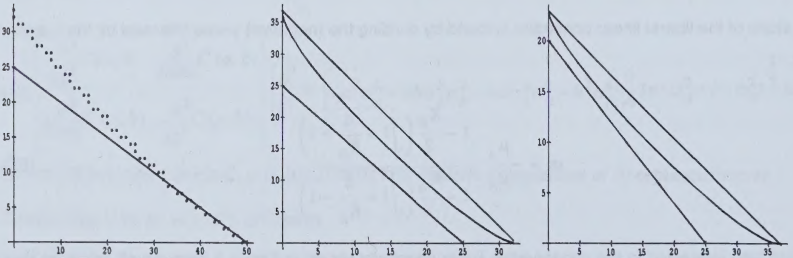
$$C_{source} = C_A + C_B \qquad \text{(EQ 13)}$$

$$C_{source} = a \cdot \mu_A + b \cdot \mu_B + \rho\sqrt{a \cdot \sigma_A^2} + \rho\sqrt{b \cdot \sigma_B^2}$$

Note that $C_{sharing} \le C_{source}$ for any given channel, service quality, operating point, and set of source types.

In allocation by individual service, we set $\tilde{C}_A$ and $\tilde{C}_B$ such that each sub-channel individually satisfies loss criterion PB_{max}. The capacity for any combination of sources will have the following linear form

$$C_{individual} = a\tilde{C}_A + b\tilde{C}_B \qquad \text{(EQ 14)}$$



A: Deterministic
B: On/Off

A: On/Off
B: Uniform

A: Uniform
B: Arbitrary

Figure 6: Resource allocation constraints for the cases of complete sharing, allocation by source type, and allocation by individual service graphed for the continued three examples. Note that for leftmost graph shows C_{souce} and $C_{sharing}$ have same constraint (true points shown) since one source is deterministic. Outside constraint is for comlete sharing, middle is allocation by source type, inside is by individual service.

VI. Effect of Burstiness on Multiplexing Gains

We find that multiplexing disciplines which allow more sharing can accommodate more services. In this section, we define a measurement of burstiness and multiplexing gains. We show that the multiplexing gains increase with burstiness.

We define the burstiness of a source to be the source rate variance divided by the source rate mean.

$$B_A = \frac{\sigma_A^2}{\mu_A} \qquad B_B = \frac{\sigma_B^2}{\mu_B} \tag{EQ 15}$$

We consider increasing the burstiness of both sources simultaneously, in proportion:

$$B = B_A = k \cdot B_B \tag{EQ 16}$$

We define C_{source} - $C_{sharing}$ to be the multiplexing gain in sharing the entire channel between all source over partitioning the channel among source types. Using EQ 12 and EQ 13, $C_{service}$ - C_{source} is shown to be an increasing function of burstiness B for fixed source rate means, operating point N, and k.

$$\begin{aligned} C_{source} - C_{sharing} &= \rho \left(\sqrt{a \cdot \sigma_A^2} + \sqrt{b \cdot \sigma_B^2} - \sqrt{a \cdot \sigma_A^2 + b \cdot \sigma_B^2}\right) \\ &= \rho \sqrt{B} \left(\sqrt{a \cdot \mu_A} + \sqrt{b \cdot k\mu_B} - \sqrt{a \cdot \mu_A + b \cdot k\mu_B}\right) \end{aligned} \tag{EQ 17}$$

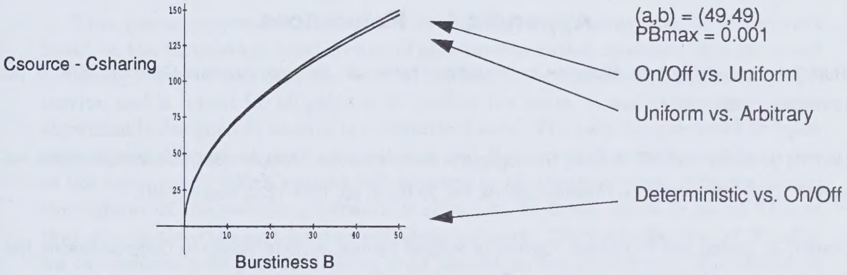


Figure 7: Graph of C_{source} - $C_{sharing}$ for parameterized examples.

We define $C_{individual}$ - C_{source} to be the multiplexing gain in sharing the channel among all services of the same type over allocating fixed bandwidth to each source. $C_{individual}$ - C_{source} is also shown to be increasing functions of burstiness B (EQ 18). We use the Chebyshev Inequality to find an upper bound on $C_{individual}$ and subtract C_{source}.

$$P(X-\mu>\varepsilon) \le P(|X-\mu|>\varepsilon) < \frac{\sigma^2}{\varepsilon^2}$$

$$P(X>\varepsilon+\mu) < \frac{\sigma^2}{\varepsilon^2}$$

$$PB_{max} = P(R_{A,n}(t) > \tilde{C}_A) < \frac{B_A\mu_A}{(\tilde{C}_A-\mu_A)^2} \qquad PB_{max} = P(R_{B,n}(t) > \tilde{C}_B) < \frac{B_B\mu_B}{(\tilde{C}_B-\mu_B)^2} \tag{EQ 18}$$

$$\tilde{C}_A < \sqrt{B_A}\sqrt{\mu_A/PB_{max}} + \mu_A \qquad \tilde{C}_B < \sqrt{B_B}\sqrt{\mu_B/PB_{max}} + \mu_B$$

$$C_{individual} - C_{source} < \left(\sqrt{B_A}\cdot\left(a\sqrt{\frac{1}{PB_{max}}} - \rho\sqrt{a}\right)\cdot\sqrt{\mu_A}\right) + \left(\sqrt{B_B}\cdot\left(b\sqrt{\frac{1}{PB_{max}}} - \rho\sqrt{b}\right)\cdot\sqrt{\mu_B}\right)$$

$$< \sqrt{B}\cdot\left(\left[a\sqrt{\frac{1}{PB_{max}}} - \rho\sqrt{a}\right]\sqrt{\mu_A} + \left[b\sqrt{\frac{1}{PB_{max}}} - \rho\sqrt{b}\right]\sqrt{\mu_B/k}\right)$$

VII. Summary

We have shown an approximate resource allocation constraint for a multiplexor with complete sharing and noted the gains over using other channel allocation schemes. Future study topics include accuracy assessment, call level control, and pricing.

Appendix 1. References

[Hui88] J.Y. Hui, "Resource Allocation for broadband networks", *Selected Areas in Commun.*, SAC-6, pp. 1598-1608, 1988.

[Jord91] S. Jordan and P.P. Varaiya, "Throughput in multiple service, multiple resource communications networks", *IEEE Transactions on Communications*, Vol. 39 No. 8, pp. 1216-1222, August 1991.

[JordPr] S. Jordan and P. Varaiya, "Control of Multiple Service, Multiple Resource Communications Networks", to appear in *IEEE Transactions on Communications.*

[Kell91] F. P. Kelly, "Effective bandwidths at multi-class queues", *Queueing Systems 9, pp. 5-15, 1991.*

[ElwaMi] Anwar I. Elwalid, Debasis Mitra, "Effective Bandwidths of General Markovian Traffic Sources and Admission Control of High Speed Networks", IEEE INFOCOM, pp. 3a.2.1-3a.2.10, 1993.

An ATM Switch with Multi-Stage Contention Resolution *

Joseph Hui and Tsern-Huei Lee*

Department of Electrical and Computer Engineering
Rutgers University
Piscataway, NJ 08855-0909
*Department of Communication Engineering
National Chiao Tung University
Hsinchu, Taiwan 300
Republic of China

Abstract

This paper proposes a large scale broadband self-routing switching network based on the three-stage construction of sort-banyan switch modules. The proposed switching network uses only one kind of module, preserves the cell sequencing of a service, and is robust for all patterns of load on the ports. A multistage three-phase algorithm is designed to control the delivery of cells. The switching network is input queued and delivers at most one cell in a cell time to each output port from one of the input ports which request cell delivery to that output port. The maximum throughput of the switching network is shown to be 0.458 which is about 78% of that of a single-stage sort-banyan switching network. With a buffer size of 20 cells, we can achieve a 40 percent loading with almost no buffer overflow. Parallelism is easily achieved by having multiple switching planes. With 4 switching planes, we show that the network is close to being output-queued.

---

*This work was done while T. H. Lee was visiting Rutgers University.

The research of J. Hui is supported by NSF under grant NSF NCR 9058079 and matching fund from Bellcore and NEC.

1 Introduction

The asynchronous transfer mode (ATM), i.e., the multiplexing of transmission capacity into short fixed length labeled transmission slots called cells, has been adopted as a flexible and efficient transmission standard for Broadband ISDN (B-ISDN). An essential component for providing B-ISDN services is a high-speed switch with a capacity ranging from hundreds of gigabits to several terabits. Many different switch fabrics such as buffered-banyan network [1], sort-banyan network [2], knockout switch [3], and memory switch [4] have been proposed recently. Due to VLSI limitations, the size of these switch fabrics cannot be arbitrarily large. Consequently, multistage interconnection of these fabrics become necessary to construct a large scale ATM switching network.

In [5], a modular three-stage switch architecture was proposed based on the generalized knockout principle. The first two stages constitute a self-routing interconnect fabric and the last stage consists of a set of output-queued switch modules. The cell loss probability can be engineered as low as desired so long as there are sufficiently many middle stage switches. However, different modules are used in different stages, meaning that more VLSI chips need to be developed. Besides, a highly bit-parallel bus is required to chain the input ports for path allocation inside the switching network. The width of the bus is equal to the number of middle-stage switches. Reliability of the bus is very crucial to achieve correct path assignments. The switch proposed in [6] uses only one switch module, i.e., the memory switch. Shared buffers are provided at the output of each switch module. Cells are transmitted stage by stage. In order to avoid the out-of-sequence problem, the path inside the switching network must be determined when a call is established. This increases not only the work to be done at call set-up but also the blocking probability of calls. It may happen that a call with bit rate v is blocked simply because no internal path has residue capacity larger than v. Also, bandwidth and path allocation for bursty calls can be difficult and perhaps inefficient.

In this paper, we propose to construct a large scale ATM switching network by interconnecting sort-banyan switch modules. The switching network we design requires only one kind of switch module, preserves the cell sequencing of a service, does not increase the internal blocking of calls, and is robust for all patterns of load on the ports.

The switch architecture and its operation are described in detail in Section 2. Section 3 analyzes the queue lengths and throughput of the switch assuming uniform traffic and binomial distribution for cell arrivals at each port. The queueing discipline is assumed to be first come first serve (FCFS) for the arrival of cells at each input port, the blocked cells will try again during the next time slot. Section 4 describes some variations, including switch parallelism, of the proposed swtith architecture. Finally, we summarize the paper in Section 5.

2 Switch Architecture and Its Operation

Fig. 1 illustrates the proposed switch architecture. The basic switch module is an $n \times n$ Batcher-banyan switch [2] and there are totally $3n$ modules. One can select $n = 32$ to construct a switch with 1024 inputs and outputs. A 4×4 Batcher-banyan switch is shown in Fig. 2 [7]. It is obvious that input ports and output ports can respectively be numbered from the top by 0 to $n^2 - 1$. Also, switch modules in each stage can be labeled respectively from the top by 0 to n-1. Furthermore, for the input ports (or output ports) connected to a particular switch module, say switch module k, one can assign numbers 0 to $n - 1$ to them from the top. As a result, input port i can be represented by (i_a, i_b) so that $i_a = [i/n]$ denotes the label of the switch module it is connected to and $i_b = i - ni_a$ denotes the number assigned within the switch module. Similarly, output port j can be represented by (j_a, j_b), with $j_a = [j/n]$ and $j_b = j - nj_a$. As usual, $[x]$ represents the largest smaller than or equal to x. We assume the switch is operated in a synchronous format so that time is divided into slots. In each slot time, the operation of the switch is divided into three phases, namely the probe phase, the ack phase, and the data phase. To describe the operation of each phase, we assume that input port i has a cell to be delivered to output port j. Also, each output port is assumed to be paired with its corresponding input port.

In phase I, the probe phase, each input port having a cell to be transmitted sends a connection request to its destination output port. To deliver connection requests, each switch module in stage one makes a random one-to-one mapping between its inputs and outputs. Consequently, each input port is connected to a randomly selected middle-stage switch module. By making such a random permutation, one can eliminate the possibility of community of interest which often results in imbalanced traffic and degrades the performance of a switch significantly. The permutation pattern can be selected a priori. Notice that the one-to-one mapping guarantees no internal blocking inside the first stage. Besides, purge and concentration (a second pass through the Batcher sorting network) can be omitted in this stage. The connection patterns in the second and the third stages are determined by the destination address. Therefore, the switching network can be self-routing if routing tags are provided. For our example, the routing tag for input port i to output port j is given by (k, j_a, j_b), where k is the label of the middle-stage switch assigned by the first-stage switch module and (j_a, j_b) represents the destination address. It is noted that internal blocking can happen in the second and the third stages. In other words, the connection request from input port i to output port j may not be granted. For convenience, we assume that this connection request survives phase I. The message contained in a connection request includes the source address, label of the assigned middle-stage switch module, and the destination address.

In phase II, the ack phase, each output port which receives a request sends an acknowledgement (ACK) back to the input port from whom the request was sent in phase I. In our example, output port j sends an ACK (via input port j, since they are paired together) back to input port i (received by output port i). The ACK will pass through the whole switch again. The routing tag for the ACK is (k, i_a, i_b). Notice that the same middle-stage switch module is used in both phase I and phase II. This is why the label of the middle-stage switch module u sed in phase I must be included in the message transmitted in that phase. Using the same middle-stage switch module for the connection request and its ACK, it is guaranteed that the ACK can always be delivered successfully, i.e., there is no internal blocking in phase II. To see this, let us consider a particular switch module in the third stage, say switch module m. Clearly, the connection requests received by the output ports connected to this switch module come from distinct middle-stage switches. Therefore, when ACKs are sent out from the corresponding input ports, which are connected to switch module m in the first stage, these ACKs go to distinct middle-stage switch modules without blocking each other. Similarly, one can argue that internal blocking cannot happen in the middle-stage because each first-stage switch module has exactly one link connected to each middle-stage switch module. An output conflict in a middle-stage switch means at least two connection requests came from the same first-stage switch module, which is impossible. Finally, no internal blocking can happen in the third stage because each input port can only issue at most one connection request in a time slot. Therefore, all the purges and concentrations can be omitted. As a result , the ACKs will pass through both the Batcher sorting network and banyan network only three times in phase II.

In phase III, the data phase, each input port which receives an ACK transmits a cell to its desired output port. In our example, input port i sends a cell to output port j. Again, no purge or concentration is required in this phase. A slot ends when phase III is completed. Figs. 3(a)-(c) show an example of the multistage three-phase algorithm for a 9×9 switching network. In this example, the request at the highest position survives when a conflict occurs. Notice that two copies of the middle-stage switch label are transmitted in phase I. The first copy serves as the routing tag of the first stage while the second copy will be received by the destined output port and used in phase II. If only one copy is transmitted, then the middle-stage switch modules have to somehow save the labels and deliver them to the destined output ports. With two copies transmitted, the first copy can be discarded after stage 1 and thus no buffering is required in the second stage.

Phase I and phase II together can be considered as an arbitration phase. One has to speed up the switching network in order to perform the arbitration phase, in addition to the data phase. The speed-up factor of the switch depends on the ratio of the time required for the arbitration phase to that for the data phase. For our design, the arbitration phase takes about $4n(n+1)+6n$ bit times for $N \times N$ switch modules where $N = n^2$. The data phase takes $53 \times 8 = 424$ bit times assuming the cell size is 53 bytes long. Therefore, the ratio is about 34% when $n = 5$.

3 Performance Analysis

In this section, we will evaluate the throughput-delay characteristics of the proposed switching network. For simplicity, it is assumed that a cell arrives at each input port with probability λ per slot. Besides, the destination of each cell is uniformly distributed over all output ports. Each input port has a buffer for incoming cells, which are served on a FCFS basis. Since an exact performance analysis based on Markov chain model is mathematically intractable for large switches, an approximate analysis is provided. The switch is modeled as N independent single server queues, one for each input port. The effect of Head of Line (HOL) blocking is taken into account by modifying the effective service time distribution.

Notice that, at the beginning of a time slot, a HOL cell can be either a blocked one which lost the previous arbitration or a fresh one which just moved to HOL. Based on the uniform traffic assumptions, the fresh HOL cells have independent destination addresses. However, the destination addresses of the blocked HOL cells are not independent because they were involved in the previous arbitrations. The number of HOL cells for the same output port j, denoted by $N_j, 0 \leq N_j \leq N-1$, affects the probability of HOL blocking. Since contention occurs only in phase I, we need only consider the blocking of connection requests.

Let us consider the N_j connection requests destined for output port j. For simplicity, we assume these N_j requests go through different middle-stage switches so that whether a particular request survives the contention in the second stage or not is independent of all the others. This assumption is close to real situation when N_j is small, which is often the case in real applications. Based on the assumption, the probability that output port j does not receive a request is equal to the probability that all the N_j requests are blocked in the second stage. Let q denote the probability that a request is blocked in the second stage. Also, let

$$\alpha(N_j) = \begin{cases} 1 & \text{at least one request survives the contention in the second stage,} \\ 0 & \text{otherwise,} \end{cases} \tag{1}$$

so that $Prob(\alpha(N_j) = 0) = q^{N_j}$ and $Prob(\alpha(N_j) = 1) = 1 - q^{N_j}$. Consequently, output port j receives a request if and only if the event $\{\alpha(N_j) = 1\}$ occures. For n is sufficiently large (e.g., $n \geq 5$), the value of q can be well approximated by

$$q = 1 - \frac{1 - e^{-\rho}}{\rho}, \tag{2}$$

where ρ is the offered load to the switch. That is, ρ is the probability that an inlet has a HOL cell at the beginning of a slot. Let $N_j^{'}$ be the value of N_j for the next slot, given by

$$N_j^{'} = N_j - \alpha(N_j) + A_j, \tag{3}$$

where A_j is the number of fresh HOL arrivals for output port j. It was shown in [8] that A_j is independent of N_j and has a Poisson density as N approaches infinity. Taking the z-transform of both sides of Eq. (3), we get

$$\begin{aligned} E[z^{N_j^{'}}] &= E[z^{N_j-\alpha(N_j)+A_j}] \\ &= E[z^{N_j-\alpha(N_j)}]E[z^{A_j}] \\ &= E\{E[z^{N_j-\alpha(N_j)} \mid N_j]\}e^{-\lambda(1-z)} \\ &= E[(1-q^{N_j})z^{N_j-1} + q^{N_j}z^{N_j}]e^{-\lambda(1-z)}. \end{aligned} \tag{4}$$

Assuming the system is at the steady state, i.e., $E[z^{N_j^{'}}] = E[z^{N_j}] = N(z)$, we get, after some manipulations,

$$N(z) = B(z)N(qz) \tag{5}$$

where

$$B(z) = \frac{z-1}{ze^{\lambda(1-z)} - 1}. \tag{6}$$

It can be verified that $B(z)$ is an increasing function of z for $z \in [0,1]$. Moreover, $N(z)$ satisfies $N(1) = 1$ and $N(q) = 1 - \lambda$. Notice that $1\text{-}N(q) = 1 - E[q^{N_j}]$ is the probability that output port j receives a cell in a time slot. Since the arrival rate must equal the departure rate at the steady state, we have $1 - N(q) = \lambda$ or $N(q) = 1 - \lambda$. To compute the mean queue length at each input port, we need to know the probability that a HOL cell is served in a slot. Assume the HOL cells are served with independent probability s. One can easily verify that $s = \lambda / E[N_j]$ at the steady state . Therefore, it remains to determine the value of $E[N_j]$, which can be obtained by computing $\left.\frac{\partial N(z)}{\partial z}\right|_{z=1}$ numerically. From Eq. (5), we have

$$\frac{\partial N(z_0)}{\partial z} = B^{'}(z_0)N(qz_0) + qB(z_0)N^{'}(z_1)\bigg|_{z_1 = qz_0}, \tag{7}$$

which implies $E[N] = B'(1)N(q) + qB(1)N'(q)$. The only unknown term in $E[N]$ is $N'(q)$ because $B(1)$ and $B'(1)$ can be computed from Eq.(6) and $N(q) = 1 - \lambda$. We can apply Eq. (7) again to get $N'(q) = B'(q)N(q^2) + qB(q)N'(q^2)$, which implies

$$E[N_j] = B'(1)N(q) + qB(1)B'(q)N(q^2) + q^2B(1)B(q)N'(q^2). \tag{8}$$

Since $N(q^2) = N(q)/B(q) = (1 - \lambda)/B(q)$ (by Eq. (5)), the only unknown term in Eq. (8) is $N'(q^2)$. Besides, the last term in Eq. (8) contains the factor q^2. If we apply Eq. (7) one more time for $N'(q^2)$, then the last term of $E[N_j]$ will contain the factor q^3. The last term will eventually become negligibly small if we apply Eq. (7) enough times. Our experience shows that, given λ and $q, E[N_j]$ can be obtained in only a few (about four) iterations. Notice that, q is a function of ρ, which can be shown is equal to $E[N_j]$ at the steady state. Therefore, to compute $E[N_j]$, we can start with a guessed initial value and compute q by Eq. (5). Using this q value, we can obtain a new $E[N_j]$. This process is repeated until it converges. The maximum throughput occurs at $\rho = E[N_j] = 1$ and is equal to 0.458.

After the values of λ and s are obtained, the mean queue length $\overline{K}$ is given by [2]

$$\overline{K} = \lambda(1 - \lambda)/(s - \lambda), \tag{9}$$

and the overflow probability is upper bounded by [2]

$$P_f \leq \lambda / s w^s, \tag{10}$$

where

$$w = \frac{\lambda(1 - s)}{(1 - \lambda)s}. \tag{11}$$

Figs. 4 and 5 show respectively the mean queue length and the upper bound of the overflow probability versus link load. In Fig. 4, the curve for mean delay of cells, denoted by $\overline{D}$, is also plotted. The mean delay is related to the mean queue length $\overline{K}$ by $\overline{D} = \overline{K}/\lambda$. In Fig. 5, B represents the buffer size at each input port. One can see that, with a buffer size of 20 cells, the overflow probability is smaller than 10^{-5} for $\lambda \leq 0.4$.

4 Variations of the Proposed Switch

The proposed switch has variations similar to those of the Batcher-banyan switch module discussed in [2]. The first variation involves the designation of cell priority. Since sorters are used in the design, cells can be provided with different priorities for contention resolution. A higher priority cell can be sorted to a favorable position when the cells with the same request for an output port are purged. One possible priority scheme is to increase the priority of a cell each time it loses arbitration so that the overflow probability of the buffer at each input port can be reduced.

The second variation is to change the three-phase algorithm into a two-phase algorithm. For this variation, there is no wire connection from the output of the Batcher sorting network to its corresponding input. Instead, a concentration network is added before the banyan network. In phase I, each input port with a cell sends the full cell through the switch. The cells which win arbitrations in the second and the third stages will go all the way to their destinations. In phase II, each output port receiving a cell acknowledges the origination of the cell by sending an ACK message to the origin. In order to guarantee non-blocking for the ACKs, the same middle-stage switches are used in phase I and phase II, just like the three-phase algorithm. Therefore, the middle-stage switch number has to be included in the cell transmitted in phase I. Besides, for this two-phase algorithm, a cell cannot be removed from its input buffer until an ACK is received.

The third variation involves the use of multiple planes in parallel as shown in Fig. 6. Effectively, the throughput capacity of the switch with S planes is equal to S times the single plane switch. Since the time required for phases I and II is about 34% of the time required for the data phase, one can easily build a two-plane switch, with the second switch plane staggering by half a slot behind the first switch plane. However, since the maximum throughput of a single switch plane is 45.8%, we suggest to construct a switch with three or four planes so that the throughput capacity is well above 1 to avoid possible congestion at input ports. Adjacent switch planes are staggered by a period of time longer than that required by phases I and II. The four-plane switch is close to being output-queued because its throughput capacity is close to 2 and the probability that more than 2 cells arrive simultaneously in a slot that are destined to the same output port is very small. For the multiple plane switch, buffers must be provided at output ports to store simultaneous arrivals from different planes. The multiple plane switch not only increases the throughput capacity but also enhances the fault tolerant capability.

5 Summary

We have proposed a large scale ATM switching network based on the three-stage construction of sort-banyan switch modules. A multistage three-phase algorithm was designed for the control of cell delivery. The proposed switching network is self-routing, preserves the cell sequencing of a service, and is robust for all patterns of load on the ports. The maximum throughput of the proposed switching network was shown to be 45.8%. One can improve the maximum throughput up to 58.6% by putting more switch modules to the middle stage to reduce the blocking probability in this stage. However, the amount of improvement in throughput may not be enough to justify the increase in complexity.

Therefore, we suggest the multiple plane switch to improve not only the performance but also the fault tolerant capability.

References

[1] J. S. Turner, "New directions in communications (or which way to the information age)," *IEEE Commun. Mag.*, vol. 24, Oct. 1986.

[2] J. Y. Hui and E. Arthurs, "A broadband packet switch for integrated transport," *IEEE J. Select. Areas Commun.*, vol. SAC-5, Oct. 1987.

[3] Y. S. Yeh, M. G. Hluchyj, and A. S. Acampora, "The Knockout switch: A simple, modular architecture for high-performance packet switching," *IEEE J. Select. Areas Commun.*, vol. SAC-5, Oct. 1987.

[4] H. Kuwahara, *et al.*, "A shared buffer memory switch for an ATM exchange," *Proc. ICC*, pp. 118-122, 1989.

[5] K. Y. Eng, M. J. Karol, and C. L. I, "A modular broadband (ATM) switch architecture with optimum performance," *Proc. ISS*, vol. 4, pp. 1-6, May 1990.

[6] Y. Sakurai, *et al.*, "Large-scale ATM multistage switching network with shared buffer memory switches," *IEEE Commun. Mag.*, vol. 29, Jan. 1991.

[7] J. Hui, "Switching integrated broadband services by sort-banyan networks," *Proc. IEEE*, vol. 79, pp.145-154, Feb. 1991.

[8] M. J. Karol, M. G. Hluchyj, and S. P. Morgan, "Input versus output queueing on a space-division packet switch," *IEEE Trans. Commun.*, vol. 35, pp. 1347-1356, Dec. 1987.

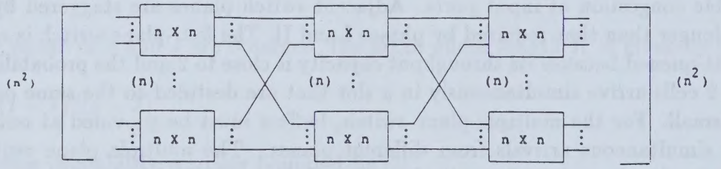


Fig. 1. A large scale switch architecture.

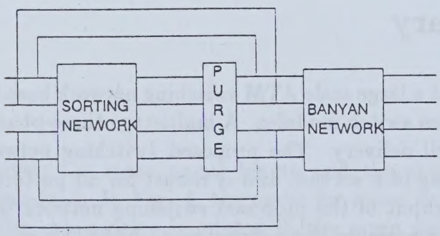


Fig. 2. A 4×4 Batcher-banyan switch module.

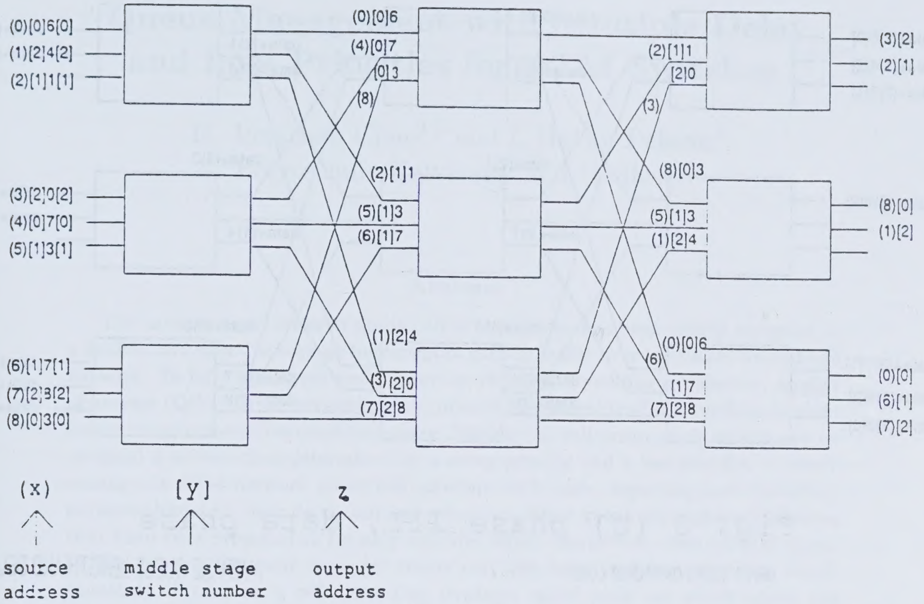

(0)[0]6[0]
(1)[2]4[2]
(2)[1]1[1]
(3)[2]0[2]
(4)[0]7[0]
(5)[1]3[1]
(6)[1]7[1]
(7)[2]8[2]
(8)[0]3[0]
(0)[0]6
(4)[0]7
[0]3
(8)
(2)[1]1
(5)[1]3
(6)[1]7
(1)[2]4
(3)
[2]0
(7)[2]8
(2)[1]1
[2]0
(3)
(8)[0]3
(5)[1]3
(1)[2]4
(0)[0]6
(6)
[1]7
(7)[2]8
(3)[2]
(2)[1]
(8)[0]
(1)[2]
(0)[0]
(6)[1]
(7)[2]
(x)
[y]
z
source address
middle stage switch number
output address

Fig. 3 (a) phase I , probe phase

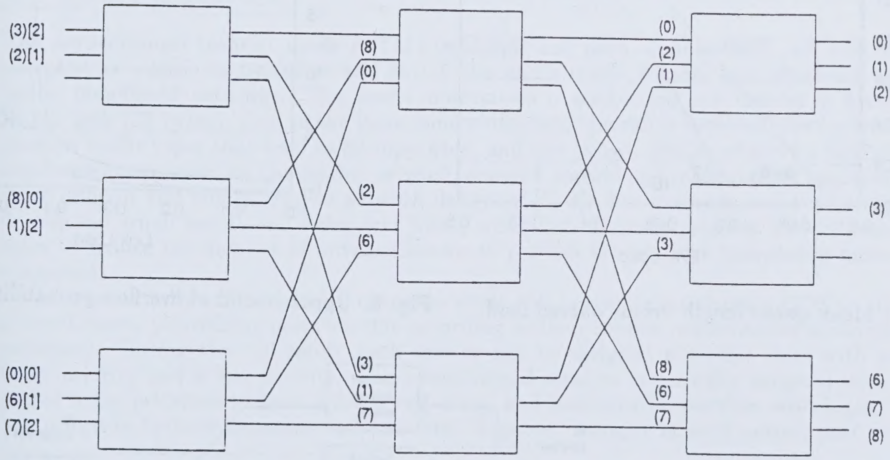

(3)[2]
(2)[1]
(8)[0]
(1)[2]
(0)[0]
(6)[1]
(7)[2]
(8)
(0)
(2)
(6)
(3)
(1)
(7)
(0)
(2)
(1)
(3)
(8)
(6)
(7)
(0)
(1)
(2)
(3)
(6)
(7)
(8)

Fig. 3 (b) phase II, ack phase

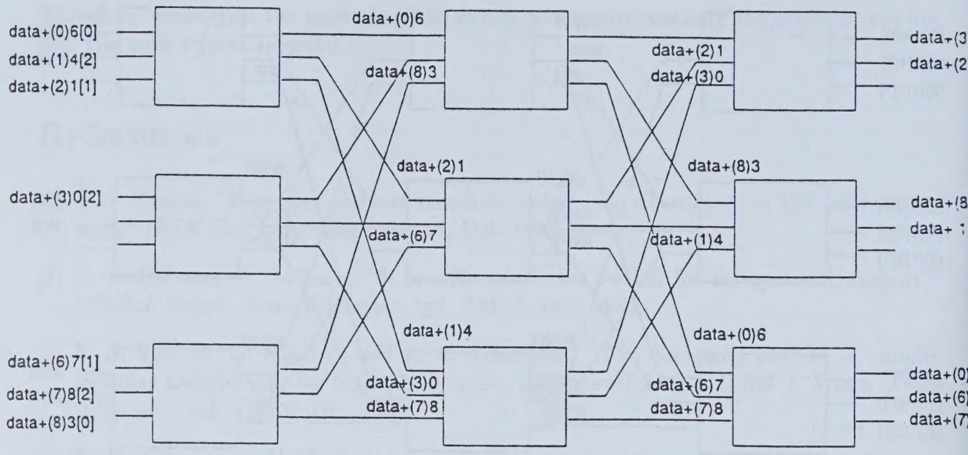

data+(0)6[0]
data+(1)4[2]
data+(2)1[1]
data+(3)0[2]
data+(6)7[1]
data+(7)8[2]
data+(8)3[0]
data+(0)6
data+(8)3
data+(2)1
data+(6)7
data+(1)4
data+(3)0
data+(7)8
data+(2)1
data+(3)0
data+(8)3
data+(1)4
data+(0)6
data+(6)7
data+(7)8
data+(3
data+(2
data+(8
data+ 1
data+(0
data+(6
data+(7

Fig. 3 (c) phase III, data phase

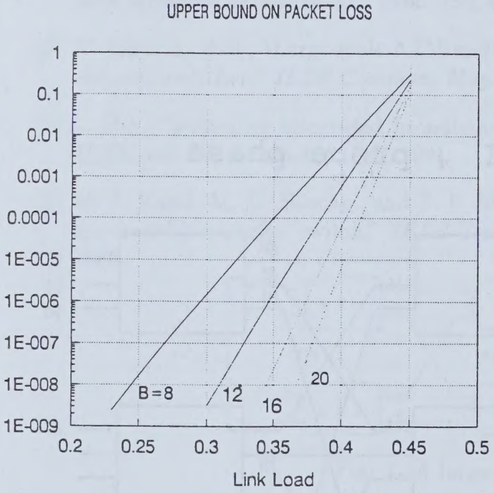


Fig. 4. Mean queue length versus offered load.

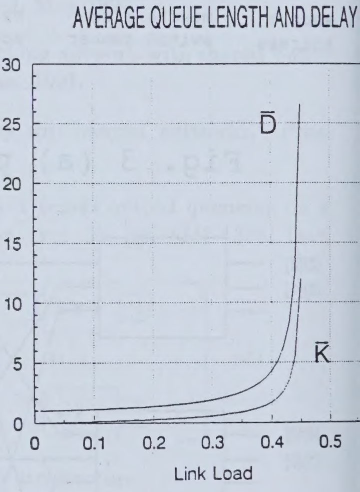


Fig. 5. Upper bound of overflow probability v

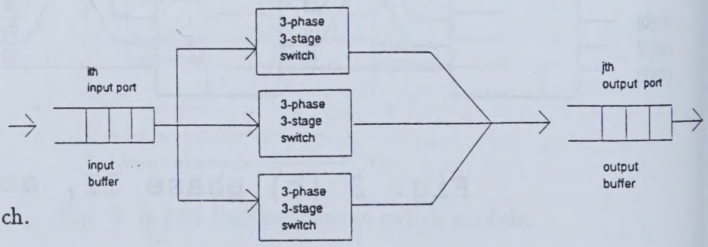

ith
input port
input
buffer
3-phase
3-stage
switch
3-phase
3-stage
switch
3-phase
3-stage
switch
jth
output port
output
buffer

Fig. 6. A three-plane switch.

Queue Management with Multiple Delay and Loss Priorities for ATM Switches *

H. Jonathan Chao[1,2] and I. Hakan Pekcan[1]
[1]Polytechnic University[†]and [2]Bellcore

Abstract

The asynchronous transfer mode (ATM) technique has been widely accepted as a flexible and effective scheme to transport various traffic over the future broadband network. To fully utilize network resources while still providing satisfactory quality of service (QOS) to all network users, prioritizing users' traffic according to their service requirements becomes necessary. During the call setup, each service can be assigned a service class determined by a delay priority and a loss priority. A queue manager in ATM network nodes will schedule ATM cells' departing and discarding sequence based on their delay and loss priorities. Most queue management schemes that have been proposed so far only consider either one of these two priority types. In this paper, the queue manager treats multiple delay and loss priorities simultaneously. Moreover, a cell discarding strategy, called push out which allows the buffer to be completely shared by all service classes, has been adopted in the queue manager. We propose a practical architecture to implement the queue manager, facilitated by a new VLSI chip, an enhanced version of the existing Sequencer VLSI chip.

1 Introduction

The asynchronous transfer mode (ATM) technique has been standardized and widely accepted as a basis to transport and switch the user's traffic flexibly and efficiently in future broadband networks. The user's information is packetized and carried in fixed length *cells* (53 bytes). Due to the large bandwidth-delay products involved, the heterogeneous traffic types that need to be supported, and the various quality of service (QOS) requirements imposed on the gigabit network, network control and management becomes more difficult and challenging to network designers [1, 2]. For example, real-time traffic requires small end-to-end delay and jitter, while data transfer requires low cell loss rates to reduce the number of retransmissions to prevent the network from being more congested.

In order to fully utilize network resources while still providing satisfactory QOS to all network users, prioritizing users's traffic according to their service requirements becomes necessary. During the call setup, each service can be assigned a service class with a delay priority and a loss priority. Time-constrained services are usually assigned with higher delay priorities to have a lower cell delay, and loss-sensitive services with higher loss priorities to have a smaller cell loss rate. A queue manager at each output port of

---

*This research is supported by Air Force Rome Lab. grant F30602-93-C-0093.

[†] Contact information: Professor H. Jonathan Chao, Department of Electrical Engineering, Polytechnic University, Six Metrotech Center, Brooklyn, New York 11201. E-mail: chao@antioch.poly.edu

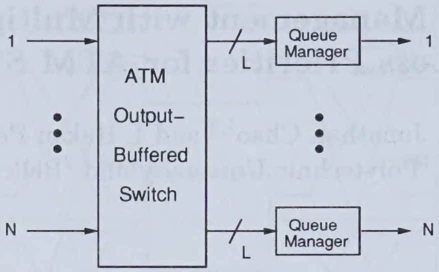


Figure 1: Cell scheduling and discarding by the queue manager at each output port of an ATM switch.

an ATM output-buffered switch, as shown in Fig. 1, is responsible for scheduling the departing and discarding sequences of ATM cells: cells with higher-delay-priority will be transmitted first, and cells with higher-loss-priority will be discarded last when the buffer is full. Note that the assignment of the delay and loss priorities to each service can be completely independent. For instance, voice and interactive video services need high delay priorities but low loss priorities because they can tolerate higher cell loss rates without noticeable imperfection. Conversely, data traffic requires high loss priorities but lower delay priorities.

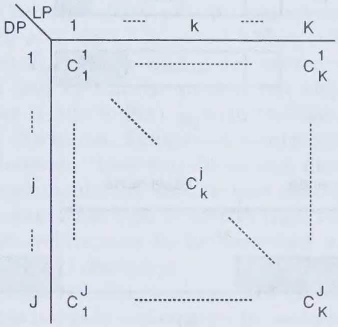
Most queue management schemes that have been proposed so far either consider one of these two priority types solely, or consider the so-called *absolute priority* [3], which provides the most preferential treatment of both delay and cell loss rates to the high-priority services. Proposed schemes that consider only the delay priority by scheduling ATM cells' departuring sequence include *fair queueing* [4, 5], *VirtualClock* [6], and *golden ratio scheduling* [7], and the schemes that only deal with the loss priority include References [8, 9]. Performance studies in Ref. [3] showed that several orders of magnitude improvement in the cell loss probability can be achieved for the high-priority services with little impact on the low-priority service performance. To our best knowledge, the queue manager described in this paper is the first one that handles both delay and loss priorities simultaneously.

Two discarding strategies, complete buffer sharing with push out and partial buffer sharing with nested thresholds for different loss priorities [10], have been proposed. In the push out strategy, an arriving high-priority cell can overwrite the low-priority cell if the buffer is filled. In the threshold strategy, a threshold is assigned to each loss priority. Cells of a particular class are accepted to enter the buffer only if the current queue length is smaller than the associated threshold. In this paper, we have adopted the push out discarding strategy for the queue manager. The reasons are explained below. First, the push out allows the buffer to be fully shared by all service classes and thus has higher throughput than the threshold does. Secondly, it is difficult to find a set of discarding thresholds for various loss-priority services. Although an optimal set could be found for given expected discarding cost (a performance penalty) [10], the thresholds will change as the traffic load and characteristics (burstiness or correlation) change. Since the push out scheme does not require the threshold setting, it does not need to know any traffic information. Thirdly, the implementation architecture proposed here can accommodate hundreds of delay or loss priorities, and may not be more complex than the nested-threshold scheme since there is no need of bookkeeping the numbers of queued cells for each loss priority as required in the threshold scheme.

In Section 2, we first study, through computer simulations, the performance in cell loss, cell delay, and the standard deviation of the cell delay for the queue manager, which deals with multiple delay and loss priorities. We evaluate two discarding schemes: complete buffer sharing with push out and partial buffer sharing with thresholds. In Section 3, we describe a practical architecture to implement the queue manager, which is improved from our previous architecture [11] to eliminate some restrictions (see Section 2.1). Section 4 gives the conclusions.

2 Performance Study for the Queue Manager

During the call setup, each service could be assigned a service class C_k^j with a delay priority (DP) equal to j and a loss priority (LP) equal to k in a priority matrix as shown in Fig. 2. Let us assume that the smaller the priority value (DP or LP) is, the higher the priority level is, i.e., C_{k-1}^{j-1} has both higher DP and LP than C_k^j. Services that require smaller delay and jitter values but permit higher cell loss rates could be assigned, for instance, the class C_K^1. On the other hand, for data traffic that has opposite QOS requirement, it could be assigned the class C_1^J. For the case of *absolute priority*, service classes only exist along the diagonal in the priority matrix. In this Section, we evaluate the performance in cell loss, queueing delay, and its standard deviation for each service class at the output queue of an ATM output-buffered switch.



C_k^j represents a service class with the delay priority (DP) j and the loss priority (LP) k.

Figure 2: Service class assignment in the priority matrix for each service.

2.1 Simulation Set-up and Traffic Model

Four different service classes, I, II, III, and IV, are considered in the simulations and are arranged in a priority matrix as shown in Fig. 3(a) (i.e., assuming $K = J = 2$ in Fig. 2). The cell departing sequence among the classes is assumed to be (I = II) → (III = IV) and the cell discarding sequence (II = IV) → (I = III), where the '=' sign means the same treatment (departing or discarding) is applied to both service classes while the '→' sign means the treatment of one class precedes the other's. In this study, classes I and II have the same delay priority and thus their cells are transmitted in the same preference. But, since they have a higher delay priority than classes III and IV do, their cells will depart earlier. In other words, cells from classes III and IV will not be transmitted until cells from classes I and II have departed. We assume that classes I and III have a higher loss priority than classes II and IV. Thus, when the buffer is filled, the cells that are from classes II and IV and queued in the buffer (if there is any) will be pushed out by those

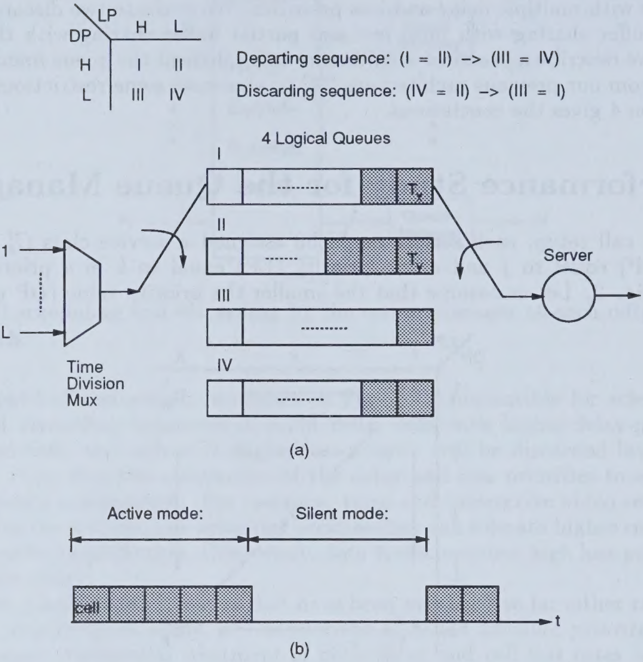


Figure 3: (a) Simulated queue manager model. (b) Cell arrival model.

from classes I and III. But, if there are no cells from classes II and IV in the buffer, the arriving cells are discarded immediately. Class II's or IV's cells will not push out any cells in the buffer.

In one of our previous work [11], we assumed the cell departing sequence among the classes as I → II → III → IV (class I preceding class II, class II preceding III, and so on) and the cell discarding sequence IV → II → III → I (class IV's cells are discarded first as the buffer is filled). For the cells that have the same DP, the higher their LPs are, the sooner they are transmitted. Similarly, for the cells that have the same LP, the lower their DPs are, the earlier they will be discarded as the buffer is full. The way of combining the DP and LP to determine the cell departing and discarding sequence allowed us to use the available VLSI Sequencer chips [12] to implement the queue manager. The chip was implemented to realize an ATM traffic shaper [13] and a queue manager using the VirtualClock mechanism [14]. However, the combination of the DP and LP has a restriction that no multiple loss priorities for each virtual connection are permitted to avoid cells being transmitted in wrong order. Here, we consider the queue management in which the departing sequence solely depends on the DP and not on the LP, and the discarding sequence only depends on the LP and not on the DP. In order to implement this new queue manager, we need to redesign a VLSI chip to enhance the existing Sequencer chip's functions.

The simulated model shown in Fig. 3(a) represents a queue manager of an ATM output-buffered switch in Fig. 1, where up to L cells may be routed through the switch and arrive simultaneously at an output port. It has been proven that, with uncorrelated traffic from input ports uniformly directed to all output ports, the probability of more than 12 cells destined for any particular output port in each cell time slot is very low (about 10^{-10}) [15]. In our simulations, we assume that there are 12 inputs and that traffic from 12 inputs to the queue manager is uncorrelated and following the active-idle pattern. We assume service class I's traffic arrives at inputs 1, 5 and 9, II's traffic at inputs 2, 6, and 10, III's traffic at inputs 3, 7, and 11, and IV's traffic at inputs 4, 8, and 12. The inputs are scanned from the top (1) to the bottom (12). In every time slot up to 12 cells are distributed to four different logical queues according to their service classes. With the complete buffer sharing strategy, the total queue length of the four logical queues is always less than or equal to the physical buffer size. A fixed-rate server chooses and sends out a cell according to cells' departing sequence provided that the buffer is not empty.

In the simulations, a time stamp is generated for each arrived cell and stored in a logical queue. The time stamp allows us to calculate the average queueing delay and its standard deviation. As shown in Fig. 3(a), T_x and T_y, are the time stamps of cells x and y. Cells that have the same delay priority are served in a first-come-first-served discipline. For instance, cell x of class I is served before cell y of class II if T_x is smaller than T_y. If T_x is equal to T_y, cells x and y are served randomly. As the buffer is filled, cells from classes II and IV can be pushed out either with the first-come-first-drop (FCFD, the oldest one in the buffer) or with the last-come-first-drop (LCFD, the youngest one in the buffer) discipline. Simulation results show that there is only a slight difference in performance between these two discarding disciplines. We have chosen the LCFD discipline for the implementation architecture described in this paper because it gives slightly better performance for high priorities than the FCFD discipline does (see Section 2.2). However, the architecture to be described in next Section can be easily modified to implement the FCFD discipline.

To quantify the traffic characteristics, we focus on a traffic model in which the input to the queue manager alternates between active and idle periods of geometrically distributed duration, as shown in Fig. 3(b). During the active period, cells arrive at the queue manager continuously in consecutive time slots. Termination of the active period occurs with probability p after each active time slot by independent Bernoulli trials. Thus, the probability that the active period (burst) will last for a duration of i time slots (consists of i cells) is

$$G(i) = p(1-p)^{i-1}, i \geq 1. \tag{1}$$

We assume that there is at least one cell in each burst. This geometric burst-length assumption yields a mean burst length of

$$R = \sum_{i=1}^{\infty} iG(i) = 1/p. \tag{2}$$

The idle period is also geometrically distributed, and termination of the idle period is also a Bernoulli process, which is characterized by another parameter q. The probability that an idle period lasts for j time slots is

$$H(j) = q(1-q)^j, j \geq 0. \tag{3}$$

Unlike the duration of an active period, we assume the duration of an idle period can be zero. The mean idle period is given by

$$S = \sum_{j=0}^{\infty} jH(j) = (1-q)/q. \tag{4}$$

Given p and q, the offered load ρ can be found by

$$\rho = \frac{R}{(S+R)}. \tag{5}$$

2.2 Simulation Results

In most of simulations, the traffic load from each service class is assumed to be 0.225, resulting in a total load of 0.9 at the queue manager's output link. Figures 4(a), (b), and (c) show the cell loss probability, queueing delay, and standard deviation of queueing delay, respectively, for four service classes versus the buffer size, ranging from 32 to 512 cells. The mean burst length of each class's traffic is assumed to be 20 cells (about 1K bytes). We also show a curve for the pure first-in-first-out (FIFO) case, where neither delay nor loss priority is considered (i.e., all four service classes are treated equally). Figure 4(a) shows the order of the cell loss rates for the four service classes is I < III < II < IV. Cell loss rates among these four classes are roughly divided into two bands (I and III in one, and II and IV in the other), corresponding to the high and low loss priorities. Although class I and III have the same cell loss priority, class I's cell loss rate is smaller than class III's because class I's cells are transmitted earlier than class III's. Similarly, class II's cell loss rate is smaller than class IV's. From Fig. 4(a), we observe that the cell loss rate of the FIFO case is between these two bands. Figures 4(b) and (c) show the queueing delay and its standard deviation are also divided into two bands (I and II in one, III and IV in the other), reflecting their high and low delay priorities. The FIFO's performance is also between these two bands.

Note that LCFD provides slightly better performance for high priorities than FCFD does. For instance, the cell loss rate of classes I and III is smaller while using the LCFD discipline than using the FCFD discipline. The reasons are explained as follows. Since classes I and II have a high delay priority than classes III and IV, among all backlogged cells in the buffer cells of classes I and II are younger than those of classes III and IV. Thus, when the LCFD discipline is applied, cells of class II will be discarded before those of classes I and III due to its lower loss priority, and also before class IV's cells because class II's cells are younger. Therefore, the LCFD discipline is unfavorable to class II in terms of cell loss rate, but favorable to other classes. Furthermore, discarding class II's cells reduces the service competition for class I because classes I and II have the same service preference. As a result, class I's average queuing delay and delay jitter is improved. On the other hand, since class III's cells push out class II's as the buffer is full, it causes a longer backlog for class III, resulting in a larger average queuing delay and delay jitter.

3 Implementation Architecture for the Queue Manager

Two separated Sequencers are used to handle both delay and loss priorities. The Sequencer that handles cells' delay priority, the Delay Priority Sequencer (DPS), simply sorts the cells such that cells with higher delay priorities are scheduled to depart earlier. The other Sequencer, Loss Priority Sequencer (LPS), arranges the cells in such a way that the cells with lower loss priorities are discarded earlier than those with higher loss priorities. Each Sequencer consists of a number of modules, and each module holds two types of information related to a cell: priority (P) and the cell's physical address (A) in the pool. Let us denote these two fields as a (P, A) pair. Note that the *departure sequence number* [14] used to implement the Virtual Clock technique can be combined with the delay priority values to form the value P, achieving fairness among the virtual channels that have the same delay priority.

As shown in Fig. 5, these pairs are stored in a Sequencer (DPS or LPS) in a descending order. Pairs with smaller P values are always located on the right of the ones with larger P values. Moreover, for pairs that have the same P values, the earlier arrivals will be placed on the right of late arrivals. As an example of inserting a new arrival into the Sequencer, let us consider an arriving cell with P=120 and $A = A_n$. This pair is broadcast to all

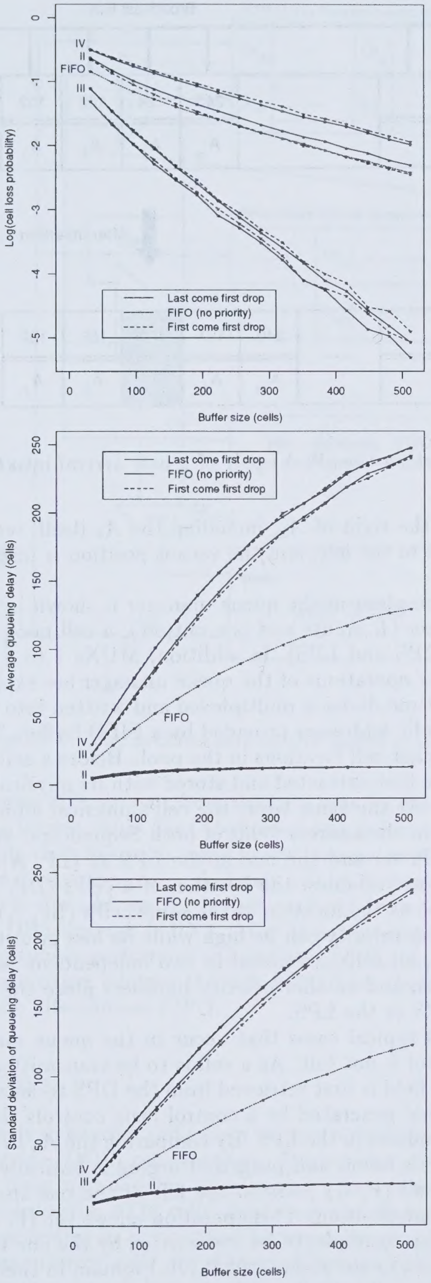


Figure 4: (a) Cell loss probability (b) Average queueing delay (c) Standard deviation of queueing delay vs. buffer size with a total offered load of 0.9 and a mean burst length of 20 cells.

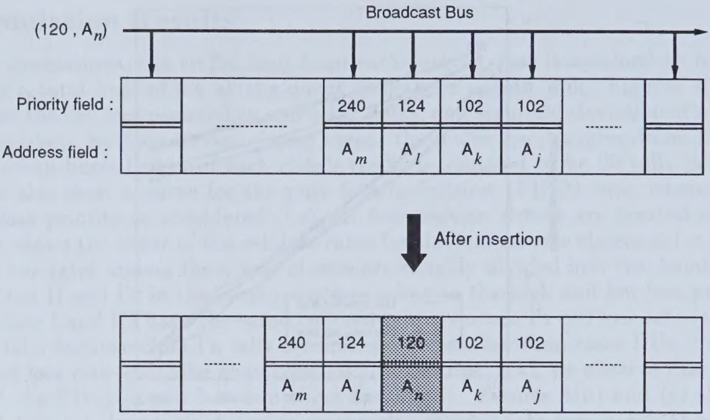


Figure 5: Inserting the (P, A) pair of a new arrival into the Sequencer.

modules. All pairs on the right of A_k, including the A_k itself, remain at their positions while others are shifted to the left, and the vacant position is inserted with the new pair $(120, A_n)$.

An architecture to implement the queue manager is shown in Fig. 6. It consists of a time-division multiplexer (L inputs and one output), a cell pool, an idle-address FIFO, and two Sequencers (DPS and LPS). In addition, MUXs 1 to 3 are used to select the data path properly. The operations of the queue manager are explained in the following. Arriving cells are first time-division multiplexed and written into the cell pool at vacant cell locations with the idle addresses provided by a FIFO buffer. The FIFO contains the addresses of current vacant cell locations in the pool. Before a cell is written into the cell pool, its DP and LP are first extracted and stored with its physical address into the DPS and LPS, respectively. At the same time, the cell's physical address, which is provided by the FIFO, is stored in the address fields of both Sequencers. We denote the resulting pair in the DPS as (DP, A) and the one in the LPS as (LP, A). Note that since DPs and LPs are independent variables, the location of a cell's (DP, A) pair in the DPS is not necessarily the same as the location of the same cell's (LP, A) pair in the LPS. This implies that a cell's delay priority can be high while its loss priority is low, or vice versa. As we indicated before, all cells are sorted in two independent Sequencers according to their DP and LP values, and smaller priority numbers place the (P, A) pairs closer to the right end of the DPS or the LPS.

Let us consider two typical cases that occur in the queue manager. The first case occurs when the cell pool is not full. As a cell is to be transmitted, e.g., (DP_n, A_n) pair in the DPS, its address field is first retrieved from the DPS to locate the cell in the pool. Meanwhile, a *purge* signal generated by a control logic controls the MUX1 to broadcast the value of A_n to all modules in the LPS. By comparing the A_n to each module's address bits, the (LP_n, A_n) pair is found and purged. Purging is basically a shift operation and is achieved as follows: all (P, A) pairs to the left of the one that is to be purged are shifted to the right by one position. This operation allows the (P, A) pair of interest, for instance (LP_n, A_n) in this example, to be overwritten by the one to the left of it. At the same time all (P, A) pairs to the right of (LP_n, A_n) remain in their locations. While the above events happen, the cell with the address A_n is read out from the cell pool and the address A_n is written back to the idle-address FIFO to let the vacant location become available for the future arriving cell.

The second case occurs when the cell pool is full (i.e., the idle-address FIFO is empty).

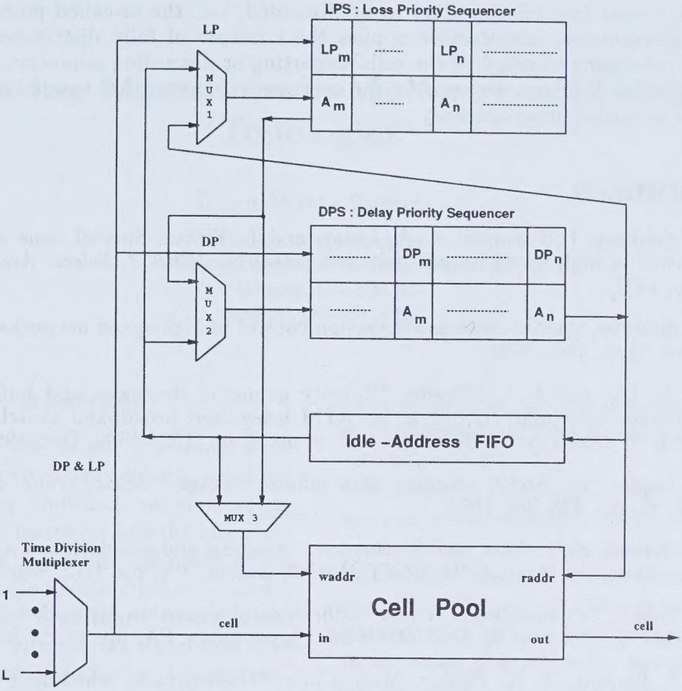

LPS : Loss Priority Sequencer
LP
LP_m
LP_n
MUX 1
A_m
A_n
DPS : Delay Priority Sequencer
DP
DP_m
DP_n
MUX 2
A_m
A_n
Idle -Address FIFO
DP & LP
MUX 3
Time Division Multiplexer
1
L
cell
waddr
raddr
Cell Pool
in
out
cell

Figure 6: The architecture of the queue manager.

In such case, the leftmost (LP, A) pair from the LPS is to be discarded. As we mentioned before, the LP with a largest value (i.e., the lowest loss priority) or the LP of the latest arrival among the cells that have the same lowest loss priority appears at the leftmost end of the LPS. For instance, the (LP_m, A_m) pair will be discarded if its LP is larger than the newly arrived one's. Its address A_m is first read out from the LPS and is broadcast to all modules of the DPS through MUX2. After the steps similar to the ones explained above, a *purging* operation is performed and the pair (DP_m, A_m) is erased. At the same time, A_m is used as a writing address through MUX3 for the newly arrived cell and is *not* written back to the idle-address FIFO.

4 Conclusions

The ATM network needs to handle traffic from various services in different ways according to their quality of service (QOS) requirements. During the call setup, each service can be assigned a service class with a delay priority and a loss priority. The queue manager in network nodes schedules the ATM cells' departing and discarding sequences based on their delay and loss priorities in a way that higher-delay-priority cells are transmitted sooner and higher-loss-priority cells are discarded last as the buffer is full. We have evaluated the performance in cell loss, queueing delay, and its standard deviation for four different service classes at the output queue of an ATM output-buffered switch. We have proposed an implementation architecture for the queue manager. The architecture allows all service classes to share the buffer until the buffer is filled up and then cells with the lowest loss priority start to be discarded, i.e., the so-called push out scheme. The implementation architecture applies the concepts of fully distributed and highly parallel processing to schedule the cells' departing or discarding sequence. An enhanced Sequencer chip has been designed for the new queue manager that handles multiple delay and loss priorities simultaneously.

References

[1] K. Sohraby, I. S. Gopal, A. A. Lazar, and L. Fratta, Special issue on congestion control in high-speed packet switched networks, *IEEE J. Select. Areas Commun.*, Sep. 1991.

[2] K. Sohraby, Special issue on congestion control in high-speed networks, *IEEE Commun. Mag.*, Oct. 1991.

[3] A. Y. Lin and J. A. Silveste, "Priority queueing strategies and buffer allocation protocols for traffic control at an ATM integrated broadband switching system," *IEEE J. Select. Areas Commun.*, vol. 9, no. 9, pp. 1524-1536, Dec. 1991.

[4] H. Nagle, "On packet switches with infinite storage," *IEEE Trans. on Commun.*, Vol. 35, pp. 435-438, 1987.

[5] A. Demers, S. Keshavt, and S. Shenker, "Analysis and simulation of a fair queueing algorithm," in *Proc. ACM SIGCOMM'89,* Austin, TX, pp. 1-12, Sept. 1989.

[6] L. Zhang, "VirtualClock: A new traffic control algorithm for packet switching networks," in *Proc. ACM SIGCOMM'90,* Philadelphia, PA, pp. 19-29, Sep. 1990.

[7] S. S. Panwar, T. K. Philips, M.-S. Chen, "Golden ratio scheduling for low delay flow control in computer networks," *IEEE Trans. on Commun.*, vol. 40, no. 4, pp. 765-772, April 1992.

[8] D. W. Petr and V. S. Frost, "Priority cell discarding for overload control in BISDN/ATM networks, " *Intl. J. of Digital and Analog Commun.*, vol. 3, no. 2, 1990.

[9] H. Kroner, G. Hebuterne, P. Boyer, and A. Gravey, "Priority management in ATM switching nodes," *IEEE J. Select. Areas Commun.*, vol. 9, no. 3, pp. 418-427, Apr. 1991.

[10] D. W. Petr and V. S. Frost, "Nested threshold cell discarding for ATM overload control: optimization under cell loss constraints," in *Proc. IEEE INFOCOM'91*, Bal Harbour, FL, April 1991.

[11] H. J. Chao and N. Uzun, "An ATM queue manager with multiple delay and loss priorities," in *IEEE GLOBECOM'92*, Orlando, FL, Dec. 1992.

[12] H. J. Chao and N. Uzun, "A VLSI Sequencer chip for ATM traffic shaper and queue manager," *IEEE J. Solid-State Circuits*, vol. 27, no. 11, pp. 1634-1643, Nov. 1992.

[13] H. J. Chao, "Design of leaky bucket access control schemes in ATM networks," in *Proc. IEEE ICC'91*, Denver, CO, pp. 180-187, June 1991.

[14] H. J. Chao, "A novel architecture for queue management in the ATM network," *IEEE J. Select. Areas Commun.*, vol. 9, no. 7, pp. 1110-1118, Sep. 1991.

[15] Y. S. Yeh, M. G. Hluchyj, and A. S. Acampora, "The knockout switch: a simple architecture for high-performance packet switching," *IEEE J. Select. Areas Commun.*, vol. 5, no. 8, pp. 1274-1283, Oct. 1987.

Routing Algorithms in Multicast Switching Topologies*

Ellen Witte Zegura
College of Computing
Georgia Institute of Technology
Atlanta, GA 30332
ewz@cc.gatech.edu

Abstract

We consider the issues of routing algorithm and switching network topology for supporting multicast communication. Key practical issues include the efficiency of the routing algorithm, the depth of the network and the blocking behavior under various loading conditions. We consider two networks with routing algorithms that insure nonblocking behavior. We also consider the effect of various routing algorithms on a network that can block, using a new analytic method for evaluating blocking probability in multicast networks.

1 Introduction

1.1 Background

Endpoints, such as telephones or computer terminals, are connected via transmission lines to switching networks to form communication systems. It is the role of the switching network to provide connections between endpoints on an as-needed basis. The value of such a capability became apparent not long after the invention of the first practical telephone. Alexander Graham Bell's telephone was patented in 1876; the first switching office was established in New Haven, Connecticut two years later [1].

While switching networks remain at the heart of communications systems, the demands being placed on their design for current and future needs are increasingly complex. The use of fiber optic transmission links has resulted in link speeds on the order of gigabits per second, placing formidable processing speed requirements on the switching network. Users of communication networks are placing challenging demands on communication system functionality. The system must be fast, of high quality, support *multipoint* connections between an arbitrary number of dynamically changing endpoints, and allow *multirate* connections such as video, audio and data to share the same network.

1.2 Paper Overview

Effective design of switching networks is critical to the success of a communication system. This paper focuses on the design of switching network *topologies* to support the needs of current and future communication systems. Topologies consist of links and crosspoints that provide interconnection between inputs and outputs. They form the basis of switching network *architectures*, which add components such as port processors, control and buffering. To reflect the practical requirements of switching systems, we are particularly interested in topologies of limited depth (since depth is proportional to delay), with little or no blocking of connections, with efficient routing algorithms and with support for multipoint and multirate connections.

---

*Part of this work was done while the author was at Washington University, St. Louis, MO. The research was supported by the NSF, Bellcore, BNR, DEC, Italtel SIT, NEC, NTT and SynOptics.

In Section 3, we consider several *nonblocking* topologies supporting multirate, multipoint connections. Informally, a nonblocking network is always able to satisfy a "legal" request. We extend the network proposed by Yang and Masson [9] to the multirate environment, constructing a network which has low asymptotic crosspoint complexity when the depth of the network is fixed. We next consider the complexity of the Yang and Masson extension and compare it to the complexity of a network proposed by Turner [7]. This reveals that while Turner's network has higher asymptotic complexity, its complexity is lower than the Yang and Masson extension for practical values of network size.

In Section 4 we consider *blocking* topologies. These networks may be unable to satisfy a legal request under certain loading conditions. This is tolerable in many application environments, particularly if it occurs infrequently and reduces the cost of the network. We use a model developed in previous work [10, 11] to evaluate blocking probability. We consider several alternative routing algorithms and compare the blocking probability under similar load across the various algorithms.

Section 2 contains formal definitions. We conclude in Section 5 with a summary of the paper and directions for additional work.

2 Definitions

For network N, we associate a quadruple (S, L, I, O), where S is a set of vertices, called *switches*, L is a set of arcs called *links*, I is a set of input terminals and O is a set of output terminals. Each link is an ordered pair (x, y) where $x \in I \cup S$ and $y \in O \cup S$. Each input and output terminal must appear in exactly one link. Links including an input terminal are called *inputs*, those containing an output terminal are called *outputs*. We will also need to refer to inputs and outputs of particular switches in the network. For $y \in S$, the *switch inputs of* y are all links (x, y). For $x \in S$, the *switch outputs of* x are all links (x, y).

The networks we consider can be divided into a sequence of *stages*, with links allowed only between switches in adjacent stages. The input vertices are in stage 0 and for $i > 0$, a vertex v is in stage i if for all links (u, v), u is in stage $i - 1$. A link (u, v) is in stage i if its left endpoint u is in stage i. We will consider only networks in which all of the outputs are in the last stage, and no other vertices are in this stage. When the outputs are in stage k, the network is called a *k-stage network.*

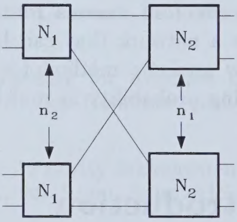


Figure 1: Series Construction $N_1 \times N_2$

The topology of many interesting networks can be described by two simple construction operators originally proposed by Cantor [3]. If N_1 is a network with n_1 outputs and N_2 is a network with n_2 inputs, then the *series connection* of N_1 with N_2 is denoted $N_1 \times N_2$ and is constructed as shown in Figure 1. Informally, this consists of taking n_2 copies of N_1 in one column and connecting them to n_1 copies of N_2 in a second column, with one link between each pair of subnetworks.

The second construction operator combines three networks. If N_1 is a network

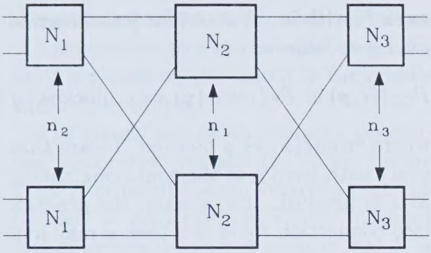


Figure 2: Parallel Construction

with n_1 outputs, N_2 is an (n_2, n_3)-network and N_3 is a network with n_1 inputs, then the *parallel connection* of N_1, N_2 and N_3 is denoted $N_1 \otimes N_2 \otimes N_3$ and is constructed as shown in Figure 2. Informally, this consists of taking n_2 copies of N_1 in one column, n_1 copies of N_2 in a second column and n_3 copies of N_3 in a third column. There is one link between each pair of subnetworks in adjacent columns. We will consider the class of networks constructed by repeated applications of the series operator and a restricted version of the parallel operator in which $N_1 = N_3 = X_{d,d}$, where $X_{d,d}$ denotes the $d \times d$ crossbar switch. We refer to such networks as *series-parallel networks*.

We illustrate the construction operators by using them to describe several popular network topologies that will be used later in the paper. The Beneš network [2], denoted $B_{n,d}$, is defined using the parallel constructor.

$$B_{d,d} = X_{d,d} \qquad B_{n,d} = X_{d,d} \otimes B_{n/d,d} \otimes X_{d,d}$$

This network has $2\log_d n - 1$ stages and $nd(2\log_d n - 1)$ crosspoints. It provides multiple paths between each input/output pair. The three stage Clos network [4], denoted $C_{n,d,m}$, is similar to the three stage Beneš network, except that the first and last stage crossbar switches do not have to be *square*, that is, have the same number of inputs and outputs.

$$C_{n,d,m} = X_{d,m} \otimes X_{n/d,n/d} \otimes X_{m,d}$$

This network has m crossbar switches in the middle stage and $2nm + md^2$ crosspoints. The number of stages can be increased to $2k+1$ for any k by repeatedly replacing the middle crossbars with three stage Clos networks.

Examples of the Beneš and Clos networks are shown in Figure 3.

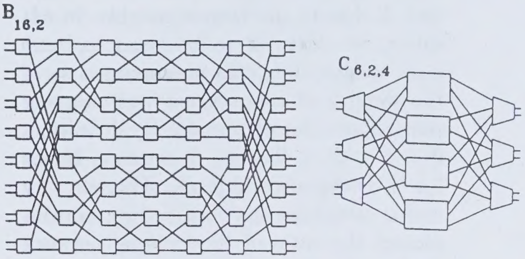


Figure 3: Beneš and Clos Networks

We are interested in *generalized connection networks* or *generalized connectors*, which operate in the multipoint environment. Generalized connectors are also called *distribution networks* or *broadcast networks* elsewhere in the literature. A *generalized connection request* is a pair (x, Y) where x is an input and Y is a nonempty set of outputs. A *generalized connection assignment* is a set of requests in which every input and output appears at most once. A *generalized connection route* is a list of links forming a tree whose root is an input and whose leaves are outputs. A route *realizes* a request (x, Y) if its root is x and its leaves are exactly the set Y. There is a second type of request in a generalized connection network. An *augmentation request* in a state s is a pair (r, y) where $r = (x, Y)$ is a request in the assignment realized by s and y is an output not in Y. An augmentation request is compatible with s if y is idle in s. An augmentation request can be satisfied in s if the route realizing r can be extended by adding unused links so that y becomes a leaf of the route.

To reflect the diversity of bandwidths required by connections, we adopt the model proposed by Melen and Turner [5] that associates a weight with each connection equal to the fraction of link bandwidth the connection requires. We let W denote the set of weights allowed in a multirate environment. For any set W, we let b denote the smallest allowable weight and B denote the largest weight. In addition we define β to be the maximum port weight, i.e., the maximum sum of the weights of connections involving any particular input or output. By definition, $0 < b \leq B \leq \beta \leq 1$. A common choice for W is the interval $[b, B]$. The quantity $1/\beta$ is called the *speed advantage*, as it indicates the ratio of the internal network speed to the external link speed. $\sigma_W(u)$ will denote the smallest $v > u$ such that v can be obtained by summing values in W.

We will consider both networks that are nonblocking and those that can block. There are several categories of nonblocking networks; the ones we consider are called *nearly nonblocking.* Informally, these networks are always able to satisfy a compatible multipoint request by routing connections according to a prescribed algorithm. In addition, all endpoints of the connection must be known at set up time. A *blocking* network reaches some states in which a compatible request cannot be realized due to insufficient resources.

In considering blocking networks, we will need some more formal definitions of blocking probability. We restrict our attention to requests of the form (x, y) where x may be busy or idle. If x is idle, the request is to create a new connection. If x is busy, the request is an augmentation request. This represents the worst case for routing, since the outputs of a multipoint connection are added one at a time.

If a compatible request is made, but cannot be satisfied, we say the request is *blocked.* We define the *generalized connection blocking probability* $P_{GC}(N, p)$ for network N with input x, output y and carried load p as follows:

$$P_{GC}(N, p) \equiv Pr\{\text{route}(x) \leftrightarrow y \text{ blocked} \mid y \text{ idle}\},$$

where "route$(x) \leftrightarrow y$ blocked" means that every path from y to the route originating at x is blocked. If x is busy, the generalized connection route is a tree; y may join the connection at any point in the tree. The input x is chosen at random from the network inputs.

3 Nonblocking Networks

A primary requirement of practical networks is that they exhibit little or no blocking of compatible requests. The next section addresses networks that can block; here we consider nonblocking networks.

3.1 Yang and Masson

In discussing multistage networks, the following terminology is used. We refer to the switches containing the inputs as *input stage* switches. We refer to the switches containing the outputs as *output stage* switches. We refer to the other switches as *middle stage* switches.

3.1.1 Three Stage Circuit Switching

In the three stage case, a generalized connection route can be specified by giving the set of middle stage switches in the route and, for each middle stage switch, the set of output stage switches it is connected to by the generalized connection route. Yang and Masson use a greedy algorithm to choose the middle stage switches used to realize a generalized connection request. They show that with "enough" middle stage switches in the network, any compatible generalized connection request can be realized with at most x middle stage switches. This has the practical effect of limiting the fanout of a connection to x in the input stage. No limit is placed on the fanout in the middle and output stages.

The greedy algorithm realizes a generalized connection request (u, Y) by finding a set R of middle stage switches that can be used to construct the route. Without loss of generality we can assume that Y is the set of output *switches* corresponding to the outputs of the generalized connection request, since multiple outputs on the same switch can be reached by branching in the output switch. The set R is initially empty. Until the switches in R can reach all output switches in Y, the following step is performed: add to R a middle stage switch accessible from u that can reach the most required output switches that are not already reached by the switches in R.

The main result of Yang and Masson's work is captured in the following theorem that expresses how many middle stage switches are needed in the network for this greedy routing approach to succeed.

Theorem 3.1 *The three stage Clos network is a nearly nonblocking generalized connector if*

$$m > \min_x\{(d-1)(x + (n/d)^{1/x})\},$$

where x is an integer in the range $1 \ldots \min\{d-1, n/d\}$.

The details of the proof are omitted; a key lemma is the following.

Lemma 3.1 *Given $(d-1)(n/d)^{1/x}$ middle switches, the greedy algorithm can find at most x of them through which to reach all of the outputs in a compatible generalized connection request (u, Y).*

For a particular choice of x it is straightforward to calculate the necessary number of middle stage switches. As is expressed in Theorem 3.1, the best choice of x is the one which minimizes the number of middle stage switches.

By choosing $x = \log(n/d)/(\delta \log\log(n/d))$ for any constant δ, $0 < \delta < 1$, it is possible to get an asymptotic bound on m of $O(d \log(n/d)/\log\log(n/d))$. Since the crosspoint count in the three stage Clos network is $2nm + md^2$, this choice of x and $d = \sqrt{n}$ yields an asymptotic crosspoint complexity of $O(n^{3/2}(\log n/\log\log n))$ for the three stage network. This is the best known explicit construction of a three stage nearly nonblocking generalized connector. The previous best such networks came from wide-sense nonblocking generalized connectors, and thus were capable of realizing augmentation requests in addition to generalized connection requests.

3.1.2 Three Stage Multirate

We now extend the Yang and Masson result to the multirate case. For more detail, see the paper by the author [8]. Consider a compatible generalized connection request (u, Y, w), where u is an input, Y is a set of outputs and $w \in W$ is an allowable connection weight. Since this connection is compatible, the weight on u is at most $\beta - w$. The same is true for each output in Y.

The results in this section are obtained by a similar approach to that used by Yang and Masson. Specifically, we first determine how many middle stage switches are needed by the greedy algorithm. We then insure that the input involved in a compatible request can reach that many middle stage switches. The key to the result is the following generalization of Lemma 3.1.

Lemma 3.2 *Given $L(w)(n/d)^{1/x}$ middle switches, the greedy algorithm can find at most x of them through which to reach all the outputs in a compatible generalized connection request, (u, Y, w), where $L(w)$ denotes the quantity $\lfloor(\beta d - w)/\sigma_W(1 - w)\rfloor$.*

With this generalization, Lemma 3.1 corresponds to the special case in which $\beta = 1$ and $W = \{1\}$, implying that $L(w) = d-1$. We use Lemma 3.2 to prove the main result.

Theorem 3.2 *The three stage Clos network is a nearly nonblocking multirate generalized connector if*

$$\begin{aligned} m &> \min_x \max_{w \in W} \{L(w)(x + (n/d)^{1/x})\} \\ &= \max_{w \in W} L(w) \min_x \{x + (n/d)^{1/x}\}, \end{aligned}$$

where x is an integer in the range $1 \ldots \min\{d-1, n/d\}$.

The specific value of $\max_{w \in W} L(w)$ will depend on the set W. One special case of interest is the *unrestricted packet switching* case in which $b = 0$ and $B = \beta$, with every value in the range $(0, B]$ allowed as a weight. In this case $\max_{w \in W} L(w)$ occurs at $w = \beta$. (Unrestricted packet switching is the worst case collection of allowable weights in the sense that increasing b or restricting W to a discrete set of weights while keeping B and β fixed will only decrease $\max_{w \in W} L(w)$.) The sufficient condition on the number of middle stage switches for unrestricted packet switching is

$$m > \left\lceil \frac{\beta(d-1)}{1-\beta} \right\rceil \min_x \{x + (n/d)^{1/x}\},$$

which is implied by

$$m > \frac{\beta(d-1)}{1-\beta} \min_x \{x + (n/d)^{1/x}\}.$$

If $(1/\beta) = 2$, this expression reduces to the same condition on m_1 derived in Theorem 3.1 for the circuit switching case. Thus with a speed advantage of two, we can handle multirate traffic with the same crosspoint complexity as for circuit switching. To put this in perspective, other multirate results indicate that a factor of two increase in complexity is about the best one can hope for in adapting a circuit switching network to accomodate multirate traffic.

3.1.3 Arbitrary Stage Networks

The approach used by Yang and Masson can be extended to an arbitrary number of stages in a straightforward way. In a $2k + 1$ stage Clos network, the greedy algorithm is used to choose the middle $2k - 1$ stage networks through which to realize a generalized connection request. This creates a generalized connection request within each of the chosen $2k - 1$ stage networks. This approach is applied recursively within each of these networks. Eventually we reach the innermost three stage Clos networks where the greedy algorithm chooses the middle stage switches through which to route the request. In effect we end up with a fanout restriction in each of the first k stages of the network.

In a similar fashion, the multirate version of the Yang and Masson network can be extended to an arbitrary number of stages. A $2k + 1$ stage Clos network requires a speed advantage of approximately $k+1$ to get crosspoint complexity equal to the $2k + 1$ stage circuit switching case.

3.2 Turner's Beneš Network

Turner has proposed a nearly nonblocking topology for circuit switching obtained by combining two Clos networks [7]. Informally, the construction puts the two networks side by side and overlaps the last stage of the "left hand" network with the first stage of the "right hand" network, giving a network of $4k+1$ stages from two $2k + 1$ stage networks.

The routing algorithm finds a switch in the overlap which has the most idle outgoing links and routes without branching from the input to that switch. All of the multipoint branching occurs in the from the overlap switch to the outputs (in the right hand network).

This approach can be extended to the multirate environment by cascading two Beneš networks [6]. The result is a nearly nonblocking generalized connector with complexity of $O(n^{1+1/(k+1)})$ and speed advantage of $1/\beta \approx k$ for $4k+1$ stages. Asymptotically, this network has higher complexity that the Yang and Masson extension. But, as the next section shows, the relative comparison for practical sizes is reversed.

3.3 Complexity Comparison

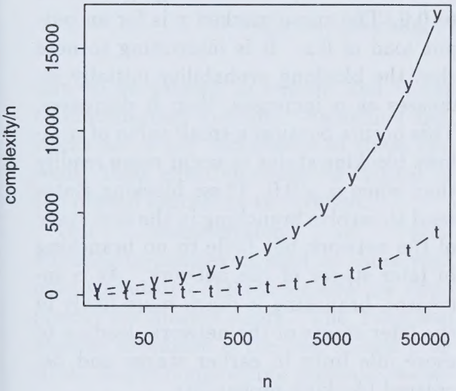


Figure 4: Five Stage Complexity

In this section we compare the complexity of the two nearly nonblocking multirate generalized connectors for five stages. While the Yang and Masson extension has lower asymptotic complexity, we are interested in comparing the complexity on practical size networks. Figure Figure 4 shows the cascaded Beneš pair labeled "t" along with the Clos networks labeled "y". The cascaded Beneš network has about one-fifth as many crosspoints as the Clos network when $n = 65536$!

4 Blocking Networks

The complexity required to achieve nonblocking capability often cannot be justified for practical systems. Rather, a guarantee of acceptably low blocking probability under reasonable loading is sufficient, particularly if the resulting network offers savings in complexity. In this section we consider several topologies and routing algorithms that can block. We look at various loading conditions and evaluate the blocking probability. This evaluation is done by means of a new analytic method for determining blocking probability in multicast networks [10, 11]. At present, the method assumes a circuit switching environment; extending it for multirate connections is a topic for future work.

4.1 Review of Blocking Model

The model we consider has two components. The first is a set of assumptions that define a probability distribution on the states of the network. The second component is equations for calculating blocking probability based on the probability distribution of the network states. The model is based on the following assumptions:

1. Every output is busy with probability p.
2. The conditions of different outputs (busy or idle) are independent.
3. If a given $d \times d$ switch has r busy outputs, the state of the switch is determined by a local fanout function $f(\cdot)$. The busy outputs are considered in random order. When an output is considered, it is connected to an arbitrary *busy* input with probability $f(u)$, where u is the current number of busy inputs. Otherwise it is connected to an arbitrary *idle* input.

We call $f(u)$ the *fanout function* since it controls the fanout structure of the multipoint connections. Using these assumptions, equations are derived for the blocking probability for the class of series-parallel networks. These equations are exact for series networks and approximate for parallel networks. The approximation gives results that are generally within about 10% of simulatation results. More detailed analysis of the accuracy is given in the previous papers.

4.2 Three Stage Clos Network

It is straightforward to adapt the equations for networks in which the crossbars have different numbers of inputs and outputs, such as the Clos network. For a three stage Clos network, we expect that the blocking probability will decrease as the number of middle stages is increased. If the fanout in the first stage is restricted, the results of Yang and Masson [9] described in the previous section indicate a sufficient value for the number of middle stages so that blocking will never occur.

Figure 5 shows the blocking probability for the Clos network $C_{16,4,m}$ as a function of output load as the number of middle stages (m) is varied. The fanout function used in stage j is $f(u) = u/r$, where r is the number of inputs to the crossbar switches in stage j. The left plot is the blocking probability for a new connection; the right plot is for augmenting an existing connection. Each curve is labeled with the value of m; the curve labeled "0" corresponds to $m = 10$. When $m = 4$, the Clos network is equivalent to the three stage Beneš network. As m is increased, the blocking probability decreases, reaching 0.0 for all values of output load when $m > 10$.

We can compare this to the results obtained by the Yang and Masson analysis. The three stage Yang and Masson result calls for $m > 12$ to get wide-sense nonblocking behavior. A better comparison would involve using the fanout function to model the restricted fanout routing of the Yang and Masson Clos network. We are looking into doing such a comparison.

4.3 Five Stage Beneš Network

We also apply the model to the five stage Beneš network. We consider a family of fanout functions of the form $f(u) = \alpha u/d$. The left plot of Figure 6 shows the blocking probability to an idle input as a function of α for output loads ranging from 0.1 to 0.9. The curve marked x is for an output load of $0.x$. It is interesting to note that the blocking probability initially increases as α increases, then it decreases. This occurs because a small value of α allows blocking states to occur more readily than when $\alpha = 0.0$. These blocking states tend to involve branching in the first stage of the network but little to no branching in later stages of the network. As α increases, branching is much more likely in the later stages of the network, leading to more idle links in earlier stages and decreased blocking probability.

For $\alpha = 0.0$, the blocking probability increases as output load increases, until very high load is reached, at which point it decreases. As the load gets very high, there is likely to be just one idle link in each stage of the network, and those idle links form a path from the (only) idle output to the (only) idle input.

The right plot of Figure 6 shows the blocking probability to a busy input as a function of α. These curves have the shape expected. The blocking probability increases as output load increases and decreases as α increases.

5 Conclusion

We have considered several routing algorithms and topologies to support the multipoint and multirate traffic which is cer-

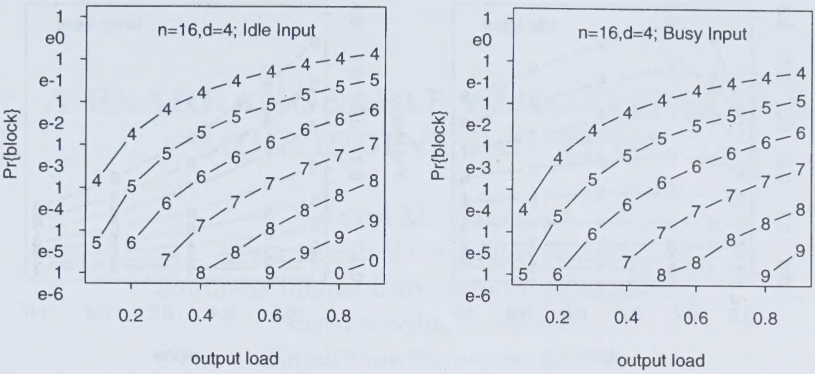


Figure 5: Blocking Probability in Clos Network

tain to be present in current and future communication systems. Switching networks are a key component of these systems; their effective design is critical to the overall success of the system.

Our consideration of non-blocking topologies focused on those with efficient routing algorithms and low complexity for limited depth. The Yang and Masson extension is the best known multirate nearly nonblocking generalized connector for fixed depth. Interestingly, the Beneš pair has substantially better complexity for practical network sizes. One must be careful to consider the constants that are hidden by an asymptotic complexity measure.

The blocking topologies offer savings in complexity at the expense of blocking. The probability of blocking is highly dependent on the system load and the structure of the multipoint connections. We are currently working on modeling more extensive routing algorithms and traffic characteristics.

References

[1] J. Bellamy. *Digital Telephony.* John Wiley & Sons, Inc., New York, second edition, 1991.

[2] V. E. Beneš. *Mathematical theory of connecting networks and telephone traffic.* Academic Press, New York, 1965.

[3] D. B. Cantor. On non-blocking switching networks. *Networks,* 1:367–377, 1971.

[4] C. Clos. A study of non-blocking switching networks. *Bell Systems Tech. Journal,* 32:406–424, 1953.

[5] R. Melen and J. S. Turner. Nonblocking multirate networks. *SIAM Journal on Computing,* 18(2):301–313, April 1989.

[6] R. Melen and J. S. Turner. Nonblocking multirate distribution networks. In *Proc. of Infocom 90,* 1990.

[7] J. S. Turner. Practical wide-sense nonblocking generalized connectors. Technical report, Washington University Computer Science Department, WUCS-88-29.

[8] E. E. Witte. The Clos network as a multirate distributor with a greedy routing algorithm. Technical report, Washington University Computer Science Department, WUCS-92-13.

[9] Y. Yang and G. M. Masson. Nonblocking broadcast switching networks. *IEEE*

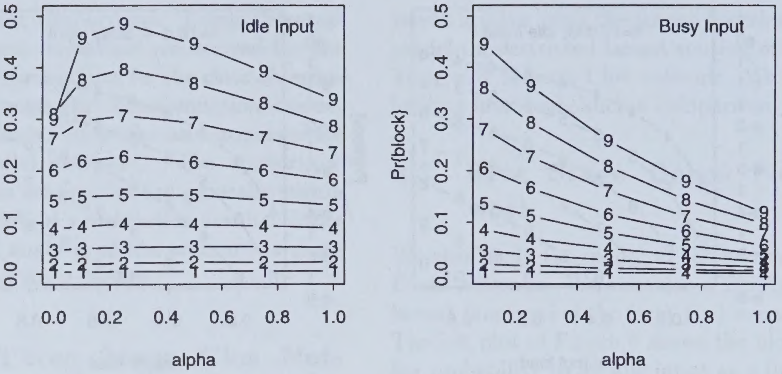


Figure 6: Blocking Probability as Alpha Varies

Transactions on Computers, 40(9):1005–1015, Sept. 1991.

[10] E. W. Zegura. An improved model for evaluating blocking probability in generalized connectors. In *submitted to Infocom 94*.

[11] E. W. Zegura. Evaluating blocking probability in distributors. In *Proc. of Infocom 93*, March 1993.

A REARRANGEABLE ALGORITHM FOR MULTIPOINT ROUTING

Bernard M. Waxman *
Department of Computer Science
Southern Illinois University at Edwardsville
Edwardsville, IL 62026
Email bmw@mpr.siue.edu

Abstract

In this paper, a rearrangeable multipoint routing scheme is considered. This represents an attempt to obtain improved performance relative to simpler, non-rearrangeable algorithms. The *rearrangeable greedy algorithm* (RGA) considered in this paper is derived from earlier work on the *weighted greedy algorithm* (WGA). As each request, either to add an endpoint or to remove an endpoint from a connection, is made, the request is first satisfied with minimum change to the connection tree. Then a single path change is allowed in order to minimize the resources used by that connection. Because of this additional change the algorithm is classified as rearrangeable.

This paper reports on a set of experiments that analyze the benefit obtained when using simple rearrangements. These experimental results show a significant reduction in cost when comparing trees generated by the rearrangeable algorithms to trees generated by the simpler WGA.

1 Introduction

The network community has shown an increased interest in multipoint (multicast) connections in conjunction with the growing importance of ATM networks. In this paper, we present the results of experiments with the *rearrangeable greedy algorithm* (RGA) for multipoint routing in communication networks. And even though this algorithm is more complex than the simpler *weighted greedy algorithm* (WGA) [8], it is still relatively simple and it lends itself to a straight forward distributed implementation. Our objective is to determine if RGA, provides enough improvement in performance to justify the additional complexity. Earlier work focused on a version of WGA that is equivalent to shortest path routing to a source (SPR). This algorithm gives "good" performance under a number of experimental tests [7, 8]. Doar and Leslie also experimented with this routing technique which they call *Naive Multicast Routing* [1]. Their results, which compare the cost of the trees generated to the cost of near optimal trees, are in close agreement with those presented in [7]. Thus, to challenge SPR, any practical multipoint algorithm must yield a

---

*This work supported by NSF grant NCR-9210548.

significant improvement in performance, and must be simple enough to allow an efficient distributed implementation in a networking environment.

Our network model is an undirected graph in which edges corresponds to network links and nodes correspond to switches. Each edge has a cost and a bandwidth. The network is connection oriented in the style of ATM networks. When a connection is first established a bandwidth requirement is specified. Only those links with sufficient free capacity to support a connection's bandwidth requirement may be used. Individual connections can be dynamic, that is, the set of endpoints or hosts may evolve over time with individual hosts being added to or removed from a connection.

Since bandwidth is reserved for each connection, communication delay is not our main concern in selecting a connection route, but instead the main criteria is the minimal use of network resources. By minimizing connection costs, we hope to maximize the use of a network's resources. The experiments presented in this paper considers connection costs as a first approximation of network resource usage. In addition, these experiments assume sufficient bandwidth on all network links. These simplifications allow us to determine if RGA justifies further consideration. The implication of the results presented here are explored in the concluding section of this paper.

The dynamic Steiner tree problem [2] serves as the basis for our multipoint routing model. Briefly, an instance of this problem consists of an undirected graph with a metric and a sequence of node requests. Each request either asks us to add a node to or remove a node from a terminal set. The set of nodes to be connected is called the terminal set, and its elements are called terminal nodes. The problem is to find a sequence of minimum cost Steiner trees that span the terminal set. Recently Westbrook and Yan have investigated the On-Line Steiner problem [9]. The On-Line problem considers sequences that consist only of add-requests. In [2], it was shown that non-rearrangeable algorithms for the On-Line problem have a worst-case lower bound of $\Omega(\log k)$ when compared to an optimum Steiner tree where k is the number of nodes in the terminal set. In [9], this result was refined to $\Omega(\log(\frac{d}{s}k))$. Here d is the maximum distance between a pair of nodes in the terminal set, and s is the cost of a minimum Steiner tree. When *remove* requests are allowed, no lower bound exists [2]. Even though these worst-case results are not very promising, the performance of non-rearrangeable algorithms is normally quite good. Also, when the Dynamic Steiner problem is modified to require that the owner remain in the connection, an algorithm such as SPR yields performance within k of optimum [10].

The remainder of this paper is organized as follows: Section 2 presents the model for dynamic connections; Section 3 presents details of the multipoint algorithm RGA; Section 4 presents performance results; and Section 5 discusses the implications of the work presented here along with a a discussion of current work involving experiments with RGA under more realistic network conditions.

2 The Connection Model

At any point in time, a multipoint connection consists of a collection of hosts, a connection tree, and a bandwidth requirement. For simplicity, the work described here assumes sufficient bandwidth is available for a connection. Thus, this work is only concerned with connection costs. Of course in real networks, the problem of bandwidth requirements can be quite complicated, and in fact may even change during the life a connection. As a first cut, we ignore the problem of bandwidth in order to determine if an algorithm justifies more complete evaluation; a procedure we have used in our evaluation of WGA.

Each host is identified with the terminal node to which it is connected. Each connection also contains a distinguished host and its terminal node, known as the *connection owner*. The owner remains during the entire lifetime of a connection, while other nodes may come and go. Each connection evolves through a sequence of requests. All requests, except for the close-connection, involve one terminal node at a time, allowing for the greatest flexibility. The four types of requests are listed below.

Open-connection An *open* request consists of a host, at a specific node, called the owner of the connection.

Add-host An *add* requests consists of host at a terminal node and the connection to which it is to be added.

Remove-host A *remove* request consists of host at a terminal node in a specific connection from which it is to be deleted.

Close-connection A *close* request consists of a connection which is to be terminated.

A remove-host request will never refer to the owner of a connection since the owner remains in the connection until a close-connection is issued. It should be noted that a connection tree will often contain nodes, along the tree paths, in addition to the terminal nodes.

In this study only one connection exists on the network at any point in time. Therefore, we measure performance by the cost of the connection trees. Since the ultimate goal is to maximize the load carried, future experiments will concentrate on performance when multiple connections may exist at any given point in time. Results presented here indicate that RGA may provide useful improvements in routing performance, and therefore justifies more extensive experiments with a more realistic network simulator.

2.1 Connection Size

In order to generate a connection of a given size, a simple probability distribution is used to determine whether to add or remove a node for a given request. The probability that a request r is an add request is given by

$$P_C(k) = \frac{\gamma(n-k)}{\gamma(n-k) + (1-\gamma)k}$$

where n is the number of nodes in the network. The value γn equals the number of hosts in a connection at equilibrium. In other words, when $k = \gamma n$, $P_C(k) = 1/2$; when $k < \gamma n$, $P_C(k) > 1/2$; and when $k > \gamma n$, $P_C(k) < 1/2$.

3 Multipoint Routing Algorithms

Based on the terminology for the dynamic Steiner tree problem, routing algorithms are classified as either non-rearrangeable or rearrangeable. Non-rearrangeable algorithms are not allowed to modify an existing connection except to add new nodes or remove unused branches of the connection. In other words, an add-host request cannot result in any link being removed from a connection, and a remove-host request cannot result in any new links being added to a connection. The rearrangeable version relaxes this restriction.

When rearrangement is allowed, it is desirable to develop algorithms where the number of rearrangements is small.

We make use of a variation of the minimum spanning tree heuristic (MST) [5], called (IMST) [7] as a measurement tool. As an approximation for the Steiner tree problem, IMST has very good average-case performance and worst-case performance within 2 of optimal [3]. Even though both IMST and MST can serve as a rearrangeable algorithms for multipoint routing, neither is a reasonable choice for real world network applications. Neither is computationally efficient for large networks; both may require extensive rearrangement; and neither lends itself to distributed implementations. In the experiments described here, we calculate the ratio of the cost of trees generated by the algorithm under investigation to the cost of the trees generated by IMST.

RGA is a collection of algorithms that is based on a simple enhancement to WGA. This modification was selected on the basis of simplicity in implementation. More complex rearrangement techniques may lead to improved performance, but may result in algorithms that are not useable for routing on real networks. After applying an initial step, that is identical to the procedure followed by WGA, a single rearrangement of the connection tree is used to reduce its cost. In those cases where such a modification does not result in a lower cost tree, no change is made. Each algorithm depends on a parameter $\omega \in [0, 0.5]$. So for example, RGA(0.3) indicates the algorithm with $\omega = 0.3$.

Informally, RGA handles a remove-host request by first pruning that portion of the connection tree which is no longer needed. In some cases, this step results in no change of the connection tree. This step is exactly the same as the procedure used by WGA. When no branch can be pruned, the algorithm checks to see if removal of the node associated with the current request will disconnect at most one terminal node. In that case, the disconnected terminal node is reconnected as if it were a new add-request. This step helps to eliminate unnecessarily long paths to single terminal nodes. To service an add-host request, RGA first joins a new node through a path to the existing connection tree $T = \{V(T),\ E(T)\}$. This path is selected to minimize a function of the distance to T and the distance from the connection point in T to the owner v_0. To add a node u to a connection, a node $v \in V(T)$ is selected to minimize the function

$$W(\omega, u, v) \;=\; (1-\omega)d(u, v) \;+\; \omega d(v, v_0)\ . \tag{1}$$

where $d(x, y)$ is the distance between a pair of nodes x and y. Then RGA selects a second path, a shortest path from the new node to a node in the connection tree that is not in the path from u to v. The cycle created is deleted by removing the longest path that still leaves a connected tree (primitive path). Note that it is possible for the primitive path to be identical to this second path. When this occurs, the rearrangement step has no effect on the resulting tree.

Figure 1 gives pseudo-code for a sequence of request $0, \ldots, k$. In the algorithm, S represents the set of terminal nodes in tree T, $\mathrm{prim}(u)$ returns the primitive path in the cycle that contains node u, and, if u has degree 2 in T, $\mathrm{neigh}_T(S, u)$ returns the two terminal nodes closest to u in T. The shortest path between x and y is returned by $\mathrm{p}(x, y)$, $V(p(x, y))$ returns the nodes in that path, and $\mathrm{p}_T(x, y)$ is the path between x and y in T. The operation $\ominus$ removes all edges and internal nodes of a path, but not the endpoints. Thus, for example, in $T \ominus p(x, y)$ the endpoints x and y are not removed.

```
begin RGA
    $T := \{\{v_0\}, \emptyset\}$;        $S := \{v_0\}$
    do $i \in \{1, 2, \ldots, k\}$ $\rightarrow$
        if request = add(u) $\rightarrow$
            connect($\omega$, $u$, $T$)
            Select $x$ yielding $\min_{x \in V(T)-V(p(u,v))} d(u, x)$
            $T := T \cup p(u, x)$
            $T := T \ominus \mathrm{prim}((u))$
        | request = remove(u) $\rightarrow$
            $S := S - \{u\}$
            do $V(T) - S$ contains a node $w$ with degree 1 $\rightarrow$
                $T := T - \{w\}$        remove w and its incident edge
            od
            if $u \in V(T)$ and $\deg_T(u) = 2$ $\rightarrow$
                Let $x$ and $y$ be $\mathrm{neigh}_T(S, u)$
                if $\deg_T(x) = 1$ $\rightarrow$
                    $T := T \ominus \mathrm{p}_T(x, y) - \{x\}$
                    connect(0.0, $x$, $T$)
                | $\deg_T(y) = 1$ $\rightarrow$
                    $T := T \ominus \mathrm{p}_T(x, y) - \{y\}$
                    connect(0.0, $y$, $T$)
                fi
            fi
        fi
    od
end RGA


begin connect($\omega$, $u$, $T$)
    Select $v$ yielding $\min_{v \in V(T)} W(\omega, u, v)$
    $T := T \cup p(u, v)$
    $S := S \cup \{u\}$
end connect
```

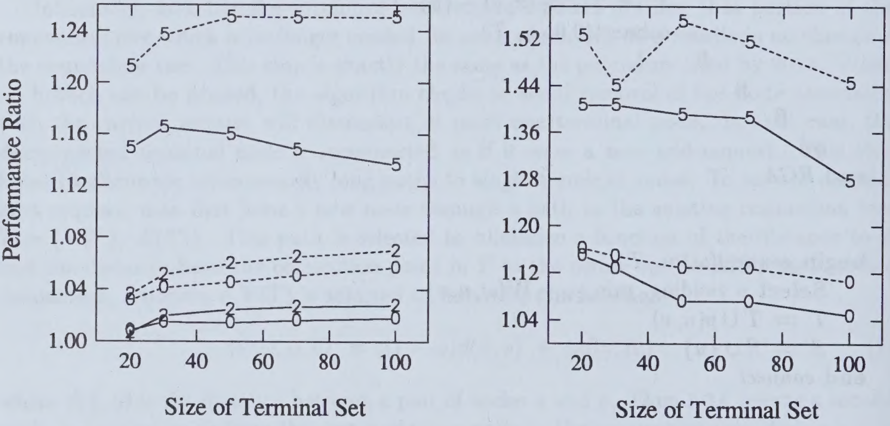
Figure 1: The Rearrangeable Greedy Algorithm (RGA).

4 Results

The simple rearrangement described in the previous section does complicate implementation in a distributed environment, but is still doable. Do we gain enough in performance to justify this additional complexity? To get a preliminary answer to this question, we have collected data from simulation experiments that measure the cost of trees constructed on both random networks and networks that have been designed by hand. These experiments compared the cost of trees generated by WGA and RGA to a near optimal solution generated by IMST.

In previous work, it has been shown that simple algorithms, such as SPR, yield relatively good performance. Experiments that compare the cost of near optimal solutions to the cost of trees generated by SPR (WGA(0.5)) are for the most part within 25% to 50% of optimal [1, 7]. Using a weight of 0.0 (greedy-routing) does somewhat better, particularly when requests only add terminal nodes. In more complex simulations, which allow for multiple connections and measure blocking rates and carried loads, SPR also yields good results [8]. It is interesting to note that the parameter ω used in WGA has much less effect in the more complex simulations than it does with the cost of a single tree. For example, there was essentially no difference in performance for SPR and greedy-routing presented in [8].

Yet as good as the performance of simpler non-rearrangeable algorithms is, there is still the potential for significant gains. The near optimal algorithms are allowed to do complete rearrangement after each request and thus, are not reasonable models for practical routing algorithms. The question then becomes how close can we approach the performance of near optimal routing with useable algorithms? The algorithms of RGA are candidates.



(a)
(b)

Figure 2: Performance of RGA on 300 node graphs restricted to add requests. Graph (a) shows average performance and (b) worst-case performance.

The results illustrated in Figure 2 compare the performance of RGA and WGA on a large random graph (300 nodes) where only add requests are allowed. Solid curves are used for RGA and the dashed curves show the results for the non-rearrangeable WGA. Each curve is labeled with the weight used, e.g., 5 indicates a value of 0.5; and each curve plots the ratio of tree costs to the cost of the solution generated by IMST. The plots in (a) show the ratio averaged over several experiments, and the plots in (b) show the ratios for the worst-case events from the same set of experiments. Note that rearrangement yields a significant improvement in performance cutting the difference in the ratio between a near optimal solution and WGA approximately in half. It is also interesting to note, that

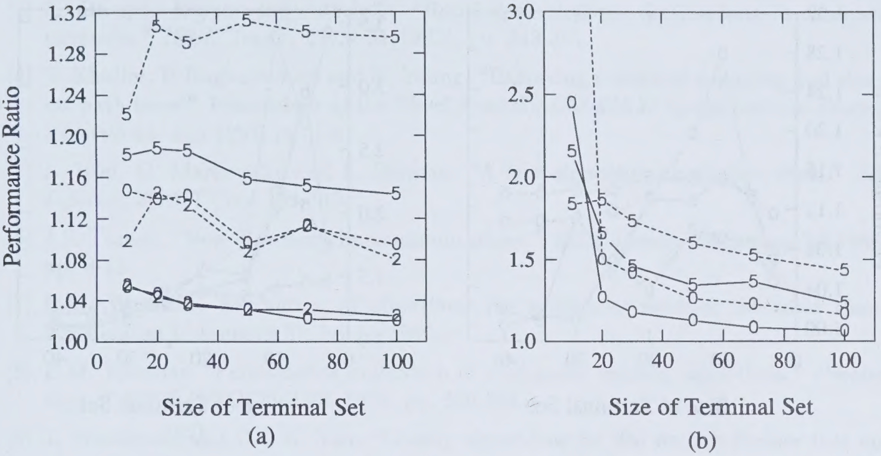


Figure 3: Performance of RGA on 300 node graphs. Graph (a) shows average performance and (b) shows worst-case performance.

in these experiments, SPR does not give performance nearly as good as greedy-routing, but when both add and remove requests are allowed the parameter ω has a much less significant effect (see for example Figure 3). The curve for $\omega = 0.2$ is not shown in (b) since it matches that of $\omega = 0.0$ very closely.

Figure 3 shows the results of another set of experiments on 300 node random graphs where both add and remove requests are allowed. The experiments make use of the probability distribution from Section 2.1 to generate trees of with the specified number of terminal nodes. Again there is a very significant gain in performance when the simple rearrangements of RGA are allowed. As shown in graph (a), routing to the owner still gives the worst performance, but the difference is not as great as in the previous set of experiments. Worst-case performance for this set of experiments is illustrated in graph (b). Note that the relative performance shown in (b) is similar to average-case results though not as consistent, and note that results for $\omega = 0.5$ do not vary as much with k. These experiments as well as earlier experiments which measured performance in terms of carried load and blocking rates indicate that for the most part a weight in the range of 0.2 to 0.3 gives the best overall performance.

In Figure 4, the results of an experiment on a 56 node graph are shown. This graph was designed by hand to represent a more realistic communication network. Results are similar to those observed for other graphs, though the differences between WGA and RGA are not as consistent as with the random graphs. In terms of worst-case performance, WGA sometimes did better than RGA. This is due in part to the smaller number of nodes in the terminal set as well as the fact that the network itself is smaller.

5 Conclusions

A new algorithm, that allows a single rearrangement after each event, has been proposed for multipoint routing. This algorithm, RGA, is simple enough that it is still practical for

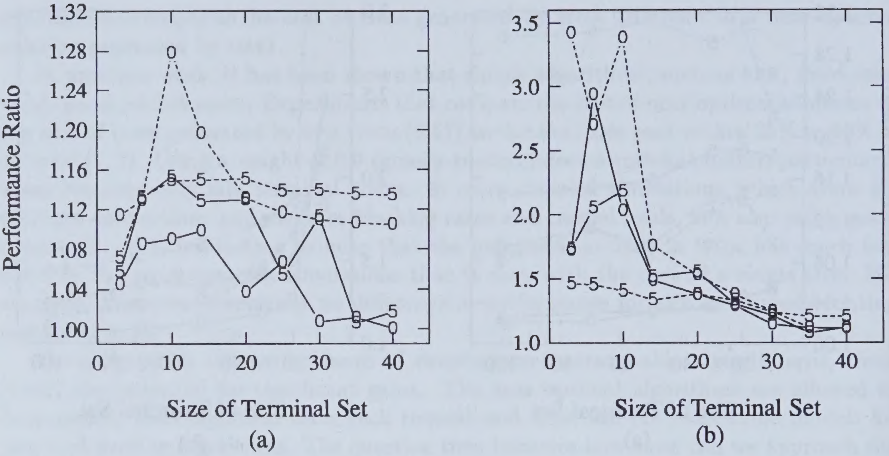


Figure 4: Performance of RGA on a hand designed 56 node graph. Graph (a) shows average performance and (b) shows worst-case performance.

use in communication networks. In addition, RGA lends itself to a distributed implementation. When measured in isolation, it generates trees with cost that is significantly lower than the corresponding non-rearrangeable WGA, reducing the difference from optimal by roughly 50%. Thus, RGA show significant promise for multipoint routing. Just to temper this optimism, the reader is reminded that differences in tree costs do not necessarily translate into significant differences in network performance. As has been indicated, the effect of varying the value of the parameter ω in WGA has significant effect on tree costs, but as reported in [8], may have little effect when used on real networks.

We are currently testing the performance of these algorithms using more realistic simulations. These simulations allow us to create multiple connections of different types on a single network. Then instead of measuring the cost of single connections we consider both blocking rates and carried load as a function of offered load. If these experiments confirm the results presented here, we plan to create and test distributed versions of RGA that will be designed to handle dynamic changes in the communication network including link and node failures. Additional factors such a quality of service need to be considered. For example, techniques like those presented by Khuller, Raghavachari, and Young in [4] to minimize the distance of terminal nodes from the owner may prove useful.

References

[1] M. Doar and I. Leslie. "How bad is Naive multicast routing?" *Proceedings of IEEE INFOCOM'93*. 1993, pp. 82-89.

[2] M. Imase and B.M. Waxman. "Dynamic Steiner tree problem." *SIAM J. on Disc. Math.*, 3(1991), pp. 369-384.

[3] K. Bharath-Kumar and J.M.Jaffe. "Routing to multiple destinations in computer networks." *IEEE Trans.*, COM-31(1983), pp. 343-351.

[4] S. Khuller, B Raghavachari and N. Young. "Balancing minimum spanning and shortest path trees." *Proceedings of the Third Annual ACM-SIAM Symposium on Discrete Algorithms.* Jan 1993, pp. 1-8.

[5] L. Kou, G. Markowsky and L. Berman. "A fast algorithm on Steiner trees." *Acta Inform.*, 3(1977), pp. 141-145.

[6] J.S. Turner. "New directions in communications." *IEEE Comm. Magazine*, 24(1986), pp. 8-15.

[7] B.M. Waxman, *Evaluation of algorithms for multipoint routing*, Doctoral thesis, Washington University St. Louis, 1989.

[8] B.M. Waxman "Performance evaluation of multipoint routing algorithms." *Proceedings of IEEE INFOCOM'93.* 1993, pp. 980-986.

[9] J. Westbrook and D.C.K. Yan. "Greedy algorithms for the on-line Steiner tree and generalized Steiner problems." *Proceedings of the Workshop on Algorithms and Data Structures, Montreal Canada.* 1993.

[10] P. Winter. "Steiner Problem in Networks: A Survey." *Networks*, 17(1987), pp. 129-167.

Computable convergence rates and geometric ergodicity for Markovian queueing systems

R.L. Tweedie
Department of Statistics
Colorado State University, Fort Collins CO 80523, USA

Abstract Many queueing systems can be represented as an irreducible Markovian model $\{\Phi_n\}$ evolving on a general space X, with n-step transition probabilities $P^n(x,A) = \mathsf{P}_x(\Phi_n \in A)$ operating on non-negative functions through $P^n(x,f) := \int P^n(x,dy)f(y)$.

We will consider situations in which convergence to an invariant measure π is geometrically fast. It can be shown [14] that for such geometrically ergodic chains a particularly strong form of convergence holds, namely for some $R < \infty$ and some $r > 1$ independent of x

$$|||P^n(x,\,\cdot\,) - \pi|||_V := \sup_x \sup_{f \le V} V(x)^{-1}|P^n(x,f) - \pi(f)| \le Rr^{-n}$$

where V is any test function satisfying the geometric drift criterion

$$P(x,V) \le \lambda V(x) + b1_C(x)$$

for some $\lambda < 1, b < \infty$ and a suitable "small set" C: it is moreover possible to show that existence of such a "drift function" V is equivalent to the existence of geometric convergence.

In this note we give an overview of recent necessary and sufficient conditions for such geometric ergodicity. We describe general bounds on the constants R, r. These are explicit in terms of λ, b and the constants defining C. In the case where the chain is stochastically monotone these results are particularly clean; in effect λ^{-1} provides a bound on r, and this is generally a best possible result.

We apply these results here to M/M/1 and M/G/1 queueing models, and develop elsewhere more wide-ranging applications.

1 Introduction to Geometric Ergodicity

As has been observed often, many queueing and network models can be analysed as continuous or discrete time Markov chains: with appropriate exponential assumptions the systems themselves are Markovian, and in much greater generality there are embedded renewal structures which can be considered in a Markovian framework, whilst the use of expanded descriptions using supplementary variables can also allow a Markov chain representation [3, 14].

For ease of exposition here we will restrict ourselves to discrete time models, but we allow our space to be quite general: this enables us to consider, in particular, much more general systems which require supplementary variables such as residual service times to render them Markovian.

Formally, we consider a Markov chain $\Phi = \{\Phi_n, n \in \mathbb{Z}_+\}$, evolving on a space X equipped with a countably generated σ-algebra $\mathcal{B}(\mathsf{X})$. The theory of such Markov chains is developed in Meyn and Tweedie [14].

The n-step transition probabilities of Φ are defined by

$$P^n(x,A) = \mathsf{P}_x(\Phi_n \in A), \qquad n \in \mathbb{Z}_+, x \in \mathsf{X}, A \in \mathcal{B}(\mathsf{X}):$$

here we use P_x to denote the measure conditional on the model starting at x, and E_x to denote the corresponding expectation. These transition kernels operate on non-negative functions through $P^n(x,f) := \int P^n(x,dy)f(y)$, and we also use $P^n f$ to denote this operation; when $n = 1$ we merely write Pf.

We will consider in this note only chains which are *φ-irreducible*: that is, for which there is a measure φ such that whenever $\varphi(A) > 0$, we have $\sum_n P^n(x,A) > 0$ for all $x \in \mathsf{X}$. We know that if the chain is φ-irreducible then [14, Chapter 5.2] any set A with $\varphi(A) > 0$ contains a *small set* C with $\varphi(C) > 0$: that is, one for which there exists some $\delta > 0$ and $n > 0$ such that

$$P^n(x,\cdot\,) \;\geq\; \delta\varphi(\,\cdot\,), \qquad x \in C. \tag{1}$$

The chain is *aperiodic* if, for some one (and then every) small set the g.c.d. of the n for which (1) holds is one [14, Chapter 5.4]. We will, in this note, largely restrict ourselves to the situation where C is a *reachable atom*: that is, a single point $\boldsymbol{\alpha}$ with $\varphi(\boldsymbol{\alpha}) > 0$, for which (1) is trivially satisfied for any n with $P^n(\boldsymbol{\alpha},\boldsymbol{\alpha}) > 0$.

It is the asymptotic behaviour of aperiodic chains, and the application of that behaviour in queueing systems, that is the focus of this review. Suppose that there exists an *invariant* probability measure π which satisfies

$$\pi(A) = \int \pi(dx)P(x,A) \qquad \text{for all } A \in \mathcal{B}(\mathsf{X}). \tag{2}$$

From [14, Chapter 13], we have immediately that in the aperiodic case, for π-a.e.initial condition $x \in \mathsf{X}$

$$\|P^n(x,\cdot\,) - \pi(\,\cdot\,)\| \to 0 \tag{3}$$

where we define the *total variation norm* of a signed measure μ by

$$\|\mu\| = \sup_{|g| \leq 1} \Big| \int \mu(dy)g(y)\Big|.$$

Indeed we know considerably more than the simple limit in (3); as shown in [14, Chapter 14], for any function $f \geq 0$ which is π-integrable we have that for π-a.e.initial condition

$$\|P^n(x,\cdot\,) - \pi\|_f := \sup_{g \leq f} |P^n(x,g) - \pi(g)| \to 0 \tag{4}$$

so that all moments of the chain converge to the appropriate limiting moment when it is finite.

In this paper we review conditions under which this convergence happens geometrically quickly, and we also describe recent bounds on the rate of such convergence. These are then compared for the embedded chain of the M/G/1 queue.

These conditions, and the associated bounds, are most usefully expressed in practice in terms of the existence of a "Foster-Lyapunov" or "drift" function V satisfying the geometric drift inequalities

$$\int P(x,dy)V(y) \leq \lambda V(x) + b\mathbb{1}_C(x) \tag{5}$$

where C is a small set, $b_C < \infty$ and $\lambda_C < 1$.

We first give several equivalent criteria for geometric ergodicity. These are taken from Chapters 15 and 16 of [14].

Theorem 1.1 *Suppose that Φ is φ-irreducible and aperiodic. Then the following are equivalent :*

(i) *There is a function $V \geq 1$, finite at least for one x, and a small set C such that for some $\lambda_C < 1$, $b_C < \infty$ the "geometric drift condition" (5) is satisfied.*

(ii) *For some small set C with $\varphi(C) > 0$, there exists $\kappa > 1$ such that*

$$\sup_{x \in C} \mathsf{E}_x[\kappa^{\tau_C}] < \infty \tag{6}$$

where τ_C is the first return time to C.

(iii) *The chain is geometrically ergodic in the sense that there is a function $V \geq 1$, finite almost everywhere (which can be taken as the V in (5)), and constants $r > 1$ and $R < \infty$ such that for all $n \in \mathbb{Z}_+$ and π-a.e x*

$$\|P^n(x, \cdot\,) - \pi\|_V \leq RV(x)r^{-n}; \tag{7}$$

and the convergence in (7) holds everywhere that V in (5) is finite.

It is valuable to note explicitly that this result shows

(i) When (5) holds for an everywhere finite function V, as it usually will in practice, then (7) and thus also (3) hold for all rather than almost all x;

(ii) The theorem covers uniform geometric convergence of moments up to "order V": that is, it ensures that for functions $g \leq V$ in (5), $\mathsf{E}_x[g(\Phi_n)] \to \mathsf{E}_\pi[g(\Phi_n)]$ in a uniform manner.

For queueing network models, V will typically take on the form $V(x) = \exp(|x|)$ for a suitable norm $|\cdot|$ on the state space [11].

Theorem 1.1 and its continuous time analogues developed in [15, 16, 13, 8] have been used in many contexts to show that chains are geometrically ergodic. These range from queueing systems such as Jackson networks [11, 7, 6] or polling systems [1, 2], to time series models [4, 18], to simulation methods such as Markov chain Monte Carlo methods [5, 10].

Typically these results are qualitative: that is, they merely show that (7) holds for some R, r but do not help to identify these constants. We now turn to some very recent results which attack this more delicate problem.

2 Bounds on Rates of Convergence

Recent results in [12] enable us to derive bounds on the rate of convergence which are actually computable. Those we describe here apply when the state space admits a reachable atom in the space: that is, essentially, a single point $\boldsymbol{\alpha}$ such that $\pi(\boldsymbol{\alpha}) > 0$. Although this is not the commonest situation, it often holds when the chain is derived from a queueing or network system, since there is a natural "empty" state which has this property.

Theorem 2.1 *Suppose that for some atom* $\alpha \in \mathcal{B}(\mathsf{X})$ *we have* $\lambda < 1, b < \infty$ *and a function* $V \geq 1$ *such that*

$$PV \leq \lambda V + b\mathbb{1}_\alpha \tag{8}$$

and also that the atom is strongly aperiodic: that is, for some δ

$$P(\alpha, \alpha) > \delta. \tag{9}$$

Then for any $r < [\vartheta(j)]^{-1}$,

$$\|P^n - \pi\|_V \leq V(x)\frac{1}{1 - r\vartheta(j)}r^{-n}, \qquad n \in \mathbb{Z}_+ \tag{10}$$

where $\vartheta(j) = 1 - M(j)^{-1}$, *and for each* $j = 1, 2, 3$

$$M(j) = \frac{1}{(1-\lambda)^2}\left[1 - \lambda + b + b^2 + \zeta_\alpha(j)(b(1-\lambda) + b^2)\right]; \tag{11}$$

here there are three possible bounds

$$\zeta_\alpha(1) \geq \zeta_\alpha(2) \geq \zeta_\alpha(3)$$

given by

$$\zeta_\alpha(1) \leq \frac{64 - 16\delta^2}{\delta^3}\left(\frac{b}{1-\lambda}\right)^2; \tag{12}$$

$$\zeta_\alpha(2) \leq \left(\frac{b}{1-\lambda}\right)^2\frac{1}{\delta p(2)} - 1 \tag{13}$$

where $p(2) = \mathsf{P}_\alpha(\tau_\alpha = 2)$; *and*

$$\zeta_\alpha(3) \leq \frac{1}{[\pi(\alpha)]^2\delta p(2)} - 1. \tag{14}$$

Proof The result using $\zeta_\alpha(1)$ is in Theorems 2.1–2.2 of [12], and requires only that we know λ, b and δ; the results using $\zeta_\alpha(2)$ or $\zeta_\alpha(3)$ follow as in the detailed calculation of the last section of [12], and require that we know at least a lower bound on $p(2)$, and for $\zeta_\alpha(3)$ that we also know $\pi(\alpha)$. □

The bounds $\zeta_\alpha(2)$ and $\zeta_\alpha(3)$ can be shown to be much smaller than $\zeta_\alpha(1)$ in general, and below we will compare them for different queuing models. It is not unrealistic in such models to assume that we have some information on the value of $p(2)$ or even $\pi(0)$ as we shall see.

For chains without an atom we can extend the results of this theorem to calculate explicit and computable bound on r; however, the statements are more complicated and we would typically expect that the queueing models we address here would admit an atom which certainly improves the accuracy of the bounds.

We next consider chains possessing a stochastic monotonicity property, for which we can improve this bound very considerably. Specifically, suppose we have a chain on $[0, \infty)$ and that $\{0\}$ is a reachable atom: and that we have the stochastic monotonicity property that for all $z \geq 0$

$$P(x, [0, z]) \geq P(y, [0, z]), \qquad x \leq y. \tag{15}$$

Then we have from [9]

Theorem 2.2 *Suppose that the chain is stochastically monotone on* $[0,\infty)$ *as in (15), and that (8) holds for the atom* $\{0\}$.

Then Φ *is geometrically ergodic, and for any* $r \le \lambda^{-1}$, *there exists a computable constant* R_r *such that*

$$\|P^n(x,\,\cdot\,) - \pi\|_V \le V(x)R_r r^{-n}, \qquad n \in \mathbb{Z}_+. \tag{16}$$

The conclusion of this theorem, namely that the rate of convergence is at least λ, also follows from recent work of Spieksma [17] under somewhat different conditions: she shows that this bound holds for chains on a countable space if the generating function

$$F_{\alpha\alpha}(z) := \sum_{n\ge1} z^n \mathsf{P}_\alpha(\tau_\alpha = n)$$

has radius of convergence at least λ^{-1} and moreover has no solutions to the equation $F_{\alpha\alpha}(z) = 1$, except $z = 1$, within the circle of radius λ^{-1}. Spieksma shows that this condition holds for a range of queueing models, but the stochastic monotone condition seems inherently easier to verify.

In Lund and Tweedie [9] it is shown that for stochastically monotone chains obeying a slightly stronger condition than (15), the actual rate of convergence is in fact the radius of convergence of $F_{\alpha\alpha}(z)$, so that in most cases this result provides not just a bound on the rate but the actual rate of convergence itself.

3 Application to M/M/1 queueing models

In this and the next section we evaluate the bounding methods above for two left continuous random walks on $\mathbb{Z}_+$, both of which are applicable to queueing models.

For a queue with exponential interarrival times of rate β and exponential service times of rate μ we can embed a Markov chain describing the numbers in the queue at times of arrival and departure. In the stable case with $\rho < 1$, where $\rho = \beta/\mu$ is the traffic intensity, this is a simple Bernouilli random walk with transition law

$$P(x, x-1) = p > 1/2, \qquad P(x, x+1) = q = 1-p, \qquad x \ge 1$$

with the boundary condition $P(0,0) = p, P(0,1) = q$. Here $p = 1/(1+\rho)$.

For this model we know [14, Chapter 15] that in (8) with $\alpha = \{0\}$ we can choose $V(x) = \gamma^x$ for some $\gamma > 1$: and for each choice of γ we get for $x \ge 1$ the value

$$\lambda_\gamma := \sum_y P(x,y)V(y)/V(x) = p\gamma^{-1} + q\gamma.$$

Choosing γ to minimize this parameter leads to

$$\gamma_{\min} = \sqrt{p/q} = 1/\sqrt{\rho}.$$

This then shows that (8) holds for $V(x) = \rho^{-x/2}$ with the corresponding parameters

$$\lambda_{\min} = \frac{2\sqrt{\rho}}{1+\rho}, \qquad b = \frac{1-\sqrt{\rho}}{1+\rho}. \tag{17}$$

We know that $\delta = p$ so we can calculate $\zeta_\alpha(1)$ in (12); in this case, using the facts that $p(2) = qp$ and that, as is well-known, $\pi(0) = 1-\rho$, we are also able to calculate $\zeta_\alpha(3)$ in (14).

In the following table we consider the results of Theorem 2.1 and Theorem 2.2 for various values of the traffic intensity ρ: here, for $j = 1,3$ the bound $\vartheta(j)$ corresponds to the different values $\zeta_\alpha(j)$ in Theorem 2.1.

ρ	$\lambda_{\min}$	$\zeta_\alpha(1)$	$\vartheta(1)$	$\zeta_\alpha(3)$	$\vartheta(3)$
0.9	0.999	155151	0.999999	761.1	0.999997
0.8	0.994	30904	0.999999	181.3	0.999949
0.7	0.984	10766	0.999998	77.0	0.999740
0.6	0.968	4656	0.999991	41.7	0.999164
0.5	0.943	2238	0.999970	26.0	0.997920
0.4	0.904	1134	0.999913	18.1	0.995628
0.3	0.843	586	0.999761	13.9	0.991955
0.2	0.745	299	0.999348	12.5	0.987150
0.1	0.575	145	0.998107	15.4	0.984254

It is immediately obvious that the actual rate of convergence, $\lambda_{\min}$, is far faster than that predicted by either of the bounds $\vartheta(j)$ that do not take into account the stochastic monotonicity; and that the use of the bound $\zeta_\alpha(2)$ is well worth the effort if the information is available.

Note that van Doorn [19] uses much more complex spectral arguments to find rates of convergence for chains such as these; it is not apparent that his methods extend to more complex models, such as those considered next.

4 Application to M/G/1 queueing models

Consider finally the numbers $\mathbf{N}^*$ in an M/G/1 queue observed immediately after each departure, where the service time has distribution H on $[0,\infty)$, and the input stream is Poisson at rate β as in the previous example. This is a Markov chain with transition matrix [14, p. 63] given by

$$P = \begin{bmatrix} q_0 & q_1 & q_2 & q_3 & q_4 & \cdots \\ q_0 & q_1 & q_2 & q_3 & q_4 & \cdots \\ & q_0 & q_1 & q_2 & q_3 & \cdots \\ & & q_0 & q_1 & q_2 & \cdots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \end{bmatrix}$$

where for each $y \geq 0$

$$q_y = \int_0^\infty \{e^{-\beta t}(\beta t)^y/y!\}\, H(dt). \tag{18}$$

For this more general chain the various quantities in the drift condition are again simple to calculate if we use the geometric function $V(x) = \gamma^x$. Let us write

$$H^*(s) = \int e^{st}\, H(dt)$$

for the Laplace transform of H, and assume that this is defined for some $s > 0$; this is known to be the appropriate condition for geometric ergodicity [14, 16.1.3]. Using (18) we have for $x > 0$

$$\begin{aligned}\lambda_\gamma &:= \textstyle\sum_0^\infty P(x,y)V(y)/V(x) \\ &= \gamma^{-1}\textstyle\sum_0^\infty q_y\gamma^y \\ &= \gamma^{-1}\textstyle\int H(dt)\exp(\beta t[\gamma-1]) \\ &= \gamma^{-1}H^*(\beta[\gamma-1])\end{aligned} \tag{19}$$

Thus, since this is a stochastically monotone chain, we have from Theorem 2.2

Proposition 4.1 *The rate of convergence of the chain* $\mathbf{N}^*$ *is given by*

$$\lambda_{\min} = \gamma_{\min}^{-1} H^*(\beta[\gamma_{\min}-1]). \tag{20}$$

where $\gamma_{\min}$ *minimises (19).* □

For any γ we have that the constant b in (8) is given, when $\boldsymbol{\alpha} = \{0\}$, by

$$b = PV(0) - \lambda_\gamma = H^*(\beta[\gamma-1]) - \lambda_\gamma = \lambda_\gamma[\gamma-1]. \tag{21}$$

We also have that $P(0,0) = q_0 = H^*(-\beta)$, so that we have all of the terms required to estimate the bounds in Theorem 2.1 also.

We will conclude by calculating some of these rates explictly when H^* has a relatively simple form, namely when H is an Erlangian distribution of order a with mean service time a/μ: this has density

$$h(t;a,\mu) = \mu^a t^{a-1}e^{-\mu t}/(a-1)! \tag{22}$$

The traffic intensity is now $\rho a\beta/\mu$; the exponential service time case corresponds to $a = 1$.

From (22) we have $H^*(s) = [\mu/(\mu-s)]^a$, and so we can find ϑ as in Proposition 4.1 by minimising

$$\lambda_x = \gamma^{-1}\Big[\frac{\mu}{\mu-\beta(\gamma-1)}\Big]^a;$$

this occurs at $\gamma_{\min} = (a+\rho)/(a+1)\rho$. The corresponding value of $\lambda_{\min}$ is given by

$$\lambda_{\min} = \Big[\frac{(a+1)\rho}{a+\rho}\Big]\Big[\frac{1+a}{a+\rho}\Big]^a. \tag{23}$$

We tabulate these values for different traffic intensities in the tables below, in the case of the exponential and the Erlangian of order $a = 2$. We also show bounds given by Theorem 2.1. These require the use of the constants b given by (21) and

$$\delta = q_0 = \Big[\frac{\mu}{\mu+\beta}\Big]^a;$$

and since for this left continuous chain we have $p(1)p(2) = q_0^2\, q_1$, we also calculate q_1 in order to use $p(2)$ in (13). For an exponential service time this is

$$\begin{aligned} q_1 &= \textstyle\int \beta t e^{-\beta t}\mu e^{-\mu t}dt \\ &= [a/\rho][\rho/(a+\rho)]^2. \end{aligned}$$

Using these we get the following values.

Parameters for convergence rates of N*;
input Poisson(β); service exponential(μ).

ρ	$\gamma_{\min}$	$\lambda_{\min}$	b	δ	$\zeta_\alpha(1)$	M_1	$\vartheta(1)$	$\zeta_\alpha(2)$	M_3	$\vartheta(2)$
0.8	1.125	0.988	0.12	0.56	34445	3789919	0.999999	1311	145223	0.999993
0.7	1.214	0.969	0.21	0.59	12766	652768	0.999998	529	27343	0.999963
0.6	1.333	0.938	0.31	0.63	5914	177529	0.999994	272	8283	0.999879
0.5	1.500	0.889	0.44	0.67	3072	61501	0.999983	161	3281	0.999695
0.4	1.750	0.816	0.61	0.71	1702	24625	0.999959	105	1562	0.999359
0.3	2.167	0.710	0.83	0.77	978	10800	0.999900	76	867	0.998846
0.2	3.000	0.556	1.11	0.83	571	5012	0.999800	63	572	0.998252
0.1	5.500	0.331	1.49	0.91	334	2400	0.999583	71	520	0.998078

It is clear that the rates for the exponential, as given by $\lambda_{\min}$, are again several orders of magnitude smaller than for ϑ, given by either of the approximations. For the Erlangian of order 2 we have:

Parameters for convergence rates of N*;
input Poisson(β); service Erlangian($2, \mu$).

ρ	$\gamma_{\min}$	$\lambda_{\min}$	b	δ	$\zeta_\alpha(1)$	M_1	$\vartheta(1)$
0.8	1.167	0.984	0.16	0.51	47124	5411808	0.999999
0.7	1.286	0.960	0.27	0.55	17040	928213	0.999998
0.6	1.444	0.922	0.41	0.59	7718	251797	0.999996
0.5	1.667	0.864	0.58	0.64	3931	87216	0.999988
0.4	2.000	0.781	0.78	0.69	2144	35032	0.999971
0.3	2.556	0.666	1.04	0.76	1218	15483	0.999935
0.2	3.667	0.507	1.35	0.83	708	7287	0.999862
0.1	7.000	0.292	1.75	0.91	415	3569	0.999719

Interestingly, the rate of convergence $\lambda_{\min}$ is faster for this chain than for the chain with simple exponential service.

Acknowledgements The author would like to express his thanks to R. Lund of the University of Georgia, and S. Meyn and D. Down of the University of Illinois for their collaboration on the results described in this paper.

Work supported in part by NSF Grant DMS-9205687.

REFERENCES

[1] E. Altman, P. Konstantopoulos, and Z. Liu. Stability, monotonicity and invariant quantities in general polling systems. *Queueing Systems*, 11:35–57, 1992.

[2] E. Altman and F. Spieksma. Polling systems–moment stability of station times and central limit theorems. Submitted, 1993.

[3] S. Asmussen. *Applied Probability and Queues.* John Wiley & Sons, New York, 1987.

[4] K. S. Chan, J. Petruccelli, H. Tong, and S. W. Woolford. A multiple threshold AR(1) model. *J. Appl. Probab.*, 22:267–279, 1985.

[5] K.S. Chan. Asymptotic behaviour of the Gibbs sampler. *J. Amer. Statist. Assoc.*, 88:320–326, 1993.

[6] D. Down and S. P. Meyn. Foster-Lyapunov criteria for stability of multi-class queueing networks. In *Proceedings of the 2nd ORSA Telecommunications Conference*, Boca Raton, FL, 1992.

[7] D. Down and S. P. Meyn. Stability of multiclass queueing networks. In *Proceedings of the 26th IEEE Conference on Information Sciences and Systems*, Princeton, NJ, 1992. Princeton University.

[8] D. Down, S. P. Meyn, and R. L. Tweedie. Geometric and uniform ergodicity of Markov processes. In preparation, 1993.

[9] R. Lund and R.L. Tweedie. Rates of convergence for pathwise ordered Markov chains. Technical report, Colorado State University, 1993.

[10] K.L. Mengersen and R.L. Tweedie. Rates of convergence in the Hastings-Metropolis algorithm. 1993. In preparation.

[11] S. P. Meyn and D. Down. Bounding delays in open Jackson networks. In *Proceedings of the 25th IEEE Conference on Information Sciences and Systems*, pages 408–416, Baltimore, MD, 1991. The Johns Hopkins University.

[12] S. P. Meyn and R. L. Tweedie. Computable bounds for convergence rates of Markov chains. Submitted, 1993.

[13] S. P. Meyn and R. L. Tweedie. Generalized resolvents and Harris recurrence of Markov processes. *Contemporary Mathematics*, 149:227–250, 1993.

[14] S. P. Meyn and R. L. Tweedie. *Markov Chains and Stochastic Stability.* Springer-Verlag, London, 1993.

[15] S. P. Meyn and R. L. Tweedie. Stability of Markovian processes II: Continuous time processes and sampled chains. *Adv. Appl. Probab.*, 1993 (to appear).

[16] S. P. Meyn and R. L. Tweedie. Stability of Markovian processes III: Foster-Lyapunov criteria for continuous time processes. *Adv. Appl. Probab.*, 1993 (to appear).

[17] F.M. Spieksma. Spectral conditions and bounds for the rate of convergence of countable Markov chains. Technical Report, University of Leiden., 1993.

[18] H. Tong. *Non-linear Time Series: A Dynamical System Approach.* Oxford University Press, Oxford, 1990.

[19] E.A. van Doorn. Conditions for exponential ergodicity and bounds for the decay parameter of a birth–death process. *Adv. Appl. Probab.*, 17:514–530, 1985.

Macroscopic approximations for a class of point processes and state-dependent non-Markovian queueing networks

Takis Konstantopoulos*
Dept. of Electrical and Computer Engineering
University of Texas
Austin, TX 78712
`takis@alea.ece.utexas.edu`

Abstract

A point process that is built by "running" a standard renewal process in a certain non-linear fashion is considered. Starting from this model, we examine convergence to a fluid limit, under suitable scaling of space and time. Deviations around the fluid limit are also considered and a functional central limit theorem is proved. The limiting process is shown to satisfy a linear time-varying stochastic differential equation driven by white noise. Subsequently, a network of queues is considered, in which service and arrival processes are modeled by point processes of the previous form. This results in a state-dependent network with Markovian routing for which fluid and diffusion approximations are proved. We use martingale methods for identifying and proving the fluid limits. We show that, unlike in the Markovian models, martingale methods are not directly applicable for proving diffusion approximations. The philosophy of the methods however is preserved by the finding of a suitable decomposition that achieves the separation between the "drift" and "diffusion" terms, which are subsequently treated separately.

1 Introduction and description of the model

Consider a standard renewal process with points $0 = T_0 < T_1 < T_2 < \cdots$. Let $\tau_i = T_{i+1} - T_i$ denote interarrival times, and let $A_t = \sum_{i=1}^{\infty} 1\{T_i \le t\}$, $t \ge 0$ denote the associated right-continuous process. A "non-linear" version of this process is obtained by defining points $T'_k =$

---

[*] Research supported by NSF grant NCR 92-11343

$\sum_{j=0}^{k-1} \tau_i/v(j)$, where $v(j)$ is a positive "speed" function. The associated counting process is $X_t = \sum_{i=1}^{\infty} 1\{T'_i \le t\}$. Note, for instance, that if $v(j)$ is equal to a constant c then $X_t = A_{ct}$.

To obtain macroscopic approximations, we consider a sequence of such processes, indexed by n, which is obtained as follows. First let $c : (0,\infty) \to (0,\infty)$ be a twice continuously differentiable function and let $v^n(j) = nc(j/n)$ be the speed of the counting process X_t^n, i.e., $X_t^n = \sum_{j=0}^{\infty} 1\{T_k^n \le t\}$, where $T_k^n = \sum_{j=0}^{k-1} \tau_j/(nc(j/n))$. Assume that the τ_j have common distribution function F with density f, mean $\lambda^{-1} > 0$ and variance σ^2, both finite. In Section 2 we show that $x_t^n := X_t^n/n$ converges, in a certain sense, to a deterministic process x_t that satisfies the differential equation $\dot{x} = \lambda c(x)$. This is done by using martingale methods. The deterministic process x is the fluid limit of the scaled point process. In Section 3 we consider the properly scaled deviation around this fluid limit, defined by $\xi_t^n = \sqrt{n}(x_t^n - x_t)$, and show that it converges, in a certain sense to be made more precise later, to a time-varying diffusion process ξ_t, given as the solution of the stochastic differential equation

$$d\xi_t = \lambda c'(x_t)\xi_t + \sqrt{\lambda^3\sigma^2 c(x_t)}dW_t \ ,$$

where W is a standard Brownian motion. We do this by finding a certain decomposition (*not* the one used in connection with martingale central limit theorems [1] for Markovian models, as, for instance, in Ethier and Kurtz [7]) that isolates the drift and diffusion parts.

Our motivation for these results arises in connection with approximation results for traffic and state processes in queueing networks. In a Markovian context, Anantharam and Konstantopoulos [1] examine functional central limit theorems for the collection of counting processes associated with the jumps of continuous time Markov chains and apply them to Jackson networks. The asymptotic covariance structure of all traffic processes within the network is thus explicitly identified. Concerning state-dependent networks, our main motivation has been the article of Kogan, Liptser and Smorodinskii [9] in which a closed network with exponential servers, state-dependent service rates and Markovian routing is considered. In Section 4 we define a network of queues with general service times and state-dependent service rates, and, under a "heavy usage" condition, we examine the functional law of large numbers and central limit theorem for suitably parametrized state processes. We thus obtain an ordinary differential equation describing the fluid limit and a stochastic differential equation describing deviations around the fluid limit. It is interesting to note that, as in our point process model, the stochastic differential equation is linear with time-varying matrix coefficients.

Recently, there has been a number of papers on deterministic and diffusion approximation of networks, and we note, for example, Chen and Mandelbaum [4]. To the best of our knowledge, however, most studies are in connection with state-independent networks, resulting in linear deterministic trajectories. Besides the aforementioned paper of Kogan et al [9], Krichagina [11] studies an exponential network with state-dependent service rates and proves in detail the functional law of large numbers by using the method of differential inclusions.

---

[1] the Doob-Meyer decomposition

2 Functional law of large numbers for the nonlinear point process

The notation and terminology is as in Section 1. Consider the natural filtration (history) of the process x_t^n, defined by $\mathcal{F}_t^n = \sigma\{x_t^n,\ 0 \leq s \leq t\}$. Then x_t^n, being a semimartingale, admits the decomposition

$$x_t^n = \tilde{x}_t^n + m_t^n \ , \tag{1}$$

where $\tilde{x}^n$ is the compensator of x^n and m^n is a martingale relative to $\mathcal{F}$. For the relevant details on compensators and martingales we refer to the comprehensive monograph of Jacod and Shiryayev [8] as well as to Brémaud [3], Liptser and Shiryayev [12], and Dacuhna-Castelle and Duflo [6]. We can explicitly compute the compensator of x^n as follows. First define

$$C_t^n = \int_0^t c(x_s^n)ds.$$

We regard the map $t \to nC_t^n$ as a time change, in that we maintain that the counting process X_t^n is related to the original renewal process A_t as

$$X_t^n = A(nC_t^n) \ .$$

To see this, it suffices to evaluate both members at $t = T_k^n$ (the k-th point of X^n). On one hand we have $X^n(T_n^k) = k$, while on the other one,

$$nC^n(T_k^n) = \int_0^{T_k^n} nc(X_s^n/n)ds = \sum_{j=0}^{k-1} \int_{T_j^n}^{T_{j+1}^n} nc(j/n)ds$$

$$= \sum_{j=0}^{k-1} nc(j/n)(T_{j+1}^n - T_j^n) = T_k \ ,$$

and also $A(T_k) = T_k$. Therefore, the compensators of x^n and A are related by $\tilde{x}_t^n = \tilde{X}_t^n/n = \tilde{A}(nC_t^n)/n$. See Brémaud [3] for the relevant theorems. To find the compensator $\tilde{A}$ of A we let $h(t) = \lim_{\epsilon \to 0} \epsilon^{-1} P(\tau_i \leq t + \epsilon | \tau_i > t) = f(t)/(1 - F(t))$ be the hazard rate of f. Also let α_t be the age of the renewal process at time t, which is the distance between t and the previously observed point: $\alpha_t = t - T_{A_t}$. One then readily obtains $\tilde{A}_t = \int_0^t h(\alpha_s)ds$. The above discussion allows us to conclude that

$$\tilde{x}^n_t = \frac{1}{n}\int_0^{nC^n_t} h(\alpha_s)ds \ .$$

We shall later also need the quadratic variation $< x^n >$ of x^n defined as the compensator of $(x^n - \tilde{x}^n)^2$. It can be seen that

$$< x^n >_t = \frac{1}{n^2}\int_0^{nC^n_t} h(\alpha_s)ds \ .$$

Let us look more closely at the asymptotic behavior of the Cesaro averages of the age process $\{\alpha_t,\ t \geq 0\}$. We have, from the standard ergodic theorem,

$$\lim_{n\to\infty} \frac{1}{n}\int_0^n h(\alpha_u)du = Eh(\hat{\alpha}) \ \text{ a.s. ,}$$

where $\hat{\alpha}$ is the stationary age; the latter has density

$$P(\hat{\alpha} \in dt) = \lambda(1 - F(t))dt \ .$$

So, $Eh(\hat{\alpha}) = \int_0^\infty h(t)\lambda(1 - F(t))dt = \lambda$. This law of large numbers can be easily translated into a functional form.

$$\sup_{0\leq t\leq T}\left|\frac{1}{n}\int_0^{nt} h(\alpha_u)du - \lambda t\right| \xrightarrow{P} 0, \quad \text{as } n \to \infty \ .$$

Indeed, the simple law of large numbers shows that the finite dimensional distributions of the process $\{Z^n_t := \frac{1}{n}\int_0^{nt} h(\alpha_u)du - \lambda t,\ t \geq 0\}$ converge to zero as $n \to \infty$. Considering Z^n as random elements of the space $\mathbf{C}[0,T]$ of continuous functions on $[0,T]$, we see that they converge to the zero function weakly. But the supremum is a continuous functional and hence the usual invariance principle, Billingsley [2], allows us to conclude.

The idea behind the proof is that in the decomposition (1) the martingale part is negligible. To guess what the deterministic limit looks like then, one need only look at the compensator of x^n. Heuristically then we see that for large n, owing to the aforementioned law of large numbers, $\tilde{x}^n_t \approx \lambda C^n_t = \lambda\int_0^t c(\tilde{x}^n_s)ds$. Thus if x^n approaches x then x should satisfy the ODE

$$\dot{x} = \lambda c(x) \ . \tag{2}$$

To make this rigorous define $\{x_t,\ t \geq 0\}$ as the solution to the ODE (2) with initial condition $x_0 = 0$. To avoid unnecessary technicalities assume that c is such that x_t does not explode in finite time. Fix a time horizon T and let $M/2$ be a bound on x_t over $0 \leq t \leq T$. Also, let L_0, L_1 be bounds on $c(x)$, $|c(x) - c(y)|/|x - y|$, respectively, over $0 < x < y < M$. We then have

$$\begin{aligned} |x^n_t - x_t| &\leq |m^n_t| + |\tilde{x}^n_t - x_t| \\ &\leq |m^n_t| + \left|\frac{1}{n}\int_0^{nC^n_t} h(\alpha_s)ds - \lambda C^n_t\right| + \lambda\int_0^t |c(x^n_s) - c(x_s)|ds \ . \end{aligned}$$

Note that we used triangle inequality by adding and subtracting the term λC_t^n. Let $D_t^n = \frac{1}{n}\int_0^{nC_t^n} h(\alpha_s)ds - \lambda C_t^n$ and use the Bellman-Gronwall lemma to obtain

$$|x_t^n - x_t| \le (|m_t^n| + |D_t^n|)e^{\lambda L_1 t} .$$

This inequality holds for all $t \le \tau(n) := \inf\{t > 0 \ : \ x_t^n > M\}$. Thus

$$\sup_{0\le t\le T\wedge\tau(n)} |x_t^n - x_t| \le (\sup_{0\le t\le T\wedge\tau(n)} |m_t^n| + \sup_{0\le t\le T\wedge\tau(n)} |D_t^n|)e^{\lambda L_1 t} .$$

The first term of the latter display is easily seen to be dominated above by $\sup_{0\le t\le L_0 T} |\frac{1}{n}\int_0^{nt} h(\alpha_u)du - \lambda t|$, which converges to zero in probability. On the other hand, for the other term we can write the estimate

$$P(\sup_{0\le t\le T\wedge\tau(n)} |m_t^n| \ge a) \le b/a^2 + P(< m^n >_{T\wedge\tau(n)}\ge b) .$$

This is Lenglart's inequality (c.f. Liptser and Shiryayev [14]). The constants a, b are arbitrary and $< m^n >_{T\wedge\tau(n)}$ converges to zero in probability, due to an earlier computation of the quadratic variation. So we proved that $\sup_{0\le t\le T\wedge\tau(n)} |x_t^n - x_t|$ converges to zero in probability. To show that $\sup_{0\le t\le T} |x_t^n - x_t|$ also converges to zero it is enough to show that $P(\tau(n) \le T) \to 0$. But $P(\tau(n) \le T) \le P(\sup_{0\le t\le T\wedge\tau(n)} |x_t^n - x_t| > M/2)$ and the same argument shows that this converges to zero as well!

3 Functional central limit theorem for the nonlinear point process

So far we considered the fluid limit x_t of the sequence of stochastic processes x_t^n, $n = 1, 2, \ldots$. We now study the deviation around x_t defined by

$$\xi_t^n = \sqrt{n}(x_t^n - x_t) .$$

As usual, we consider x^n as a random element of the space of càdlàg functions $\mathbf{D}[0,\infty)$. Had the process x_t^n been Markovian, the martingale central limit theorem, see, e.g., Liptser and Shiryayev [13], would have been directly applicable, as in the models of Kogan et al [9] and Krichagina [11]. The martingale CLT calls for a Doob-Meyer decomposition of the semimartingale ξ_t^n and requires that, loosely speaking, the compensator have no diffusion limit. It can be seen (see [10] for more details) that this is not the case here. However, having in mind that the goal is to simply isolate the drift and diffusion parts, and taking into account the time change representation $X_t^n = A(C_t^n)$, we write

$$\xi_t^n = \zeta_t^n + \eta_t^n \ ,$$

where

$$\zeta_t^n = \frac{A(nC_t^n) - \lambda n C_t^n}{\sqrt{n}} \ , \ \eta_t^n = \sqrt{n}\lambda(C_t^n - C_t) \ .$$

Recall that $C_t^n = \int_0^t c(x_s^n)ds$, and $C_t = \int_0^t c(x_s)ds$. Observe now that ζ_t^n is a time change of $Z_t^n := n^{-1/2}(A(nt) - \lambda nt)$ for which a functional CLT is available (see Billingsley [2]), namely,

$$Z^n \Rightarrow \sqrt{\lambda^3\sigma^2}W, \ \text{ as } n \to \infty \ .$$

Here W is a standard Brownian motion in $\mathbf{R}$ and $\Rightarrow$ denotes weak convergence. Omitting details, we also see that the random time change C^n converges weakly to C, and a continuity argument (see Billingsley [2], and Anantharam and Konstantopoulos [1] for a similar use of the argument) shows that ζ_t^n converges, as a random element of $\mathbf{D}[0,\infty)$, to the time-varying Brownian motion $W(C_t)$. But the latter has the same distribution as the process defined by the Itô integral

$$\widetilde{W}_t := \int_0^t \sqrt{\lambda^3\sigma^2 c(x_s)dW_s} \ .$$

The other term, η_t^n, can now be treated as follows. Write a Taylor expansion and obtain

$$\begin{aligned}\eta_t^n &= \lambda\sqrt{n}\int_0^t (c(x_s^n) - c(x_s))ds \\ &= \lambda\int_0^t c'(x_s)\xi_s^n ds + \frac{1}{2}\lambda\int_0^t c''(y_s^n)\xi_s^n(x_s^n - x_s)ds \ ,\end{aligned}$$

where y_s^n is between x_s and x_s^n. One can show, using a Bellman-Gronwall lemma again, that ξ^n is a tight sequence in $\mathbf{D}[0,\infty)$. Thus the η_t^n has the same limit as the first term of the above display.

The above discussion made us conclude that the two components, ζ_t^n, η_t^n of ξ_t^n have the same limits as $\int_0^t \sqrt{\lambda^3\sigma^2 c(x_s)}dW_s = \widetilde{W}_t$, and $\lambda\int_0^t c'(x_s)\xi_s^n ds$, respectively. Thus it is now natural to speculate that if ξ_t^n has a limit ξ_t then this limit will satisfy

$$\xi_t = \lambda\int_0^t c'(x_s)\xi_s ds + \int_0^t \sqrt{\lambda^3\sigma^2 c(x_s)}dW_s \ .$$

This can be rigorously shown as follows. Since ξ^n is tight it has at least one limit point, say $\hat{\xi}$. Consider the map $\Phi : \mathbf{D}[0,\infty) \to \mathbf{D}[0,\infty)$ defined by

$$\Phi(\vartheta)_t = \vartheta_t - \lambda\int_0^t c'(x_s)\vartheta_s ds$$

and observe that it is continuous in the Skorokhod topology. Then $\Phi(\xi^n) \Rightarrow \Phi(\hat{\xi})$. But $\Phi(\xi^n) \Rightarrow$

$\widetilde{W}$. Hence any limit point $\hat{\xi}$ satisfies $\Phi(\hat{\xi}) = \widetilde{W}$. By the uniqueness of the solution of this problem we conclude that the weak limit of ξ^n is unique and is given as the solution of the stochastic differential equation

$$d\xi_t = \lambda c'(x_t)\xi_t dt + \sqrt{\lambda^3\sigma^2 c(x_t)}dW_t \ .$$

Note that this is a linear, time-varying SDE.

4 Macroscopic approximations for state-dependent networks

Consider a network of queues defined as follows. There are K single server queues. To each server i we associate a sequence of i.i.d. random variables σ^i_m, $m = 1, 2 \ldots$ representing "service times" with mean μ_i^{-1} and variance v_i, and a sequence of i.i.d. random variables τ^i_m, $m = 1, 2, \ldots$ with mean λ_i^{-1} representing interarrival times. The model is an open or closed network of GI/GI/1 queues except that each server serves customers with a population-dependent speed. Thus at the start of the m-th service, server i uses the scaled variable $\sigma^i_m/(nc_i(X_1/n, \ldots, X_K/n))$, where $X_1, \ldots, X_K$ denote the queue lengths at that time, and $c(x_1, \ldots, x_K)$ is a positive function defined on $\mathbf{R}^K_+$, twice continuously differentiable. The parameter n is some large parameter that may be taken as proportional to the initial population.

To make things more concrete, and to avoid unnecessary notation, assume that the network is closed and irreducible. Let $X^n(t)$ be the queue length vector at time t and let $x^n(t) = X^n(t)/n$. Assume that $x^n(0) \to x(0)$, as $n \to \infty$. Then the process x^n converges to a deterministic process x defined as the solution of the ODE

$$\dot{x} = c(x)M(P - I) \ ,$$

where M is the diagonal matrix diag$(\mu_1, \ldots, \mu_K)$. We shall prove the convergence under the "heavy usage" condition that the deterministic trajectory $x(t)$ is such that it does not hit zero. The general case can be proved by using the methods results of Krichagina [11] adapted to our non-Markovian network.

To prove this convergence, we use the methods of Sections 2 and 3. First observe that as long as the process $X^n(t)$ does not hit the boundary it can be represented as

$$X^n_i(t) = X^n_i(0) + \sum_{k=1}^{K} B^n_{ki}(t) - B^n_i(t) \ , \quad i = 1, \ldots, K \ .$$

The processes B^n_i, B^n_{ij} are time changes of renewal processes, in a manner analogous to our simple model. Let A_i be a renewal process constructed by the unscaled service variables

σ^i_m, $m = 1, 2, \ldots$ of the i-th queue and let A_{ij} be obtained by sampling A_i with probabilities p_{ij}. Clearly the latter is also a renewal process with rate $\mu_i p_{ij}$. Define the time change

$$C^n_i(t) = \int_0^t c^n_i(x^n(s))ds$$

and let $B^n_i(t) := A_i(nC^n_i(t))$, $B^n_{ij}(t) := A_{ij}(nC^n_{ij}(t))$. From this representation one sees that the compensator $\tilde{x}^n$ of the process x^n, with respect to its natural history, is given by

$$\tilde{x}_i(t) = \frac{1}{n}\sum_k \tilde{A}_k(nC^n_k(t))p_{ki} - \frac{1}{n}\tilde{A}_i(nC^n_i(t)) \ .$$

This expression is valid up to a stopping time $\tau(n)$ defined as the first time that the process x^n enters a $1/n$-neighborhood of the boundary of $\mathbf{R}^K_+$, which can be shown to converge to infinity in probability. As in the Section 2, we have the functional law of large number $\tilde{A}_i(nt)/n \to \mu_i t$. Using the arguments of Section 2 we conclude that x^n has the same limit as its compensator, and the latter is seen to be a deterministic process x satisfying

$$x_i(t) = x_i(0) + \sum_k \mu_k p_{ki} \int_0^t c_k(x(s))ds - \mu_i \int_0^t c_i(x(s))ds \ .$$

This concludes the sketch of the proof.

The functional central limit theorem can also be identified as follows. Let

$$\xi^n_i(t) = \xi^n_i(0) + \sqrt{n}[x^n_i(t) - x_i(t)] \ .$$

Assume that $\xi^n_i(0)$ converges weakly to some random vector $\xi(0)$. Use the notation $C_i(t) = \int_0^t c_i(x(s))ds$ and write the decomposition

$$\xi^n_i(t) = \xi^n_i(0) + \zeta^n_i(t) + \eta^n_i(t) \ ,$$

where

$$\zeta^n_i(t) = \sum_k \frac{A_{ki}(nC^n_k(t)) - \mu_k p_{ki} n C^n_k(t)}{\sqrt{n}} - \frac{A^n_i(t) - \mu_i n C^n_i(t)}{\sqrt{n}},$$

$$\eta^n_i(t) = \sqrt{n}\sum_k \mu_k p_{ki}[C^n_k(t) - C_k(t)] - \sqrt{n}\mu_i[C^n_i(t) - C_i(t)] \ .$$

Again, this decomposition was obtained by adding and subtracting the term $\sqrt{n}\{\sum_k \mu_k p_{ki} C^n_k(t) - \mu_i C^n_i(t)\}$, and is valid up to a stopping time. Now observe that ζ^n_i is a time change of a linear combination of the collection of *dependent* renewal processes $\{A_i, A_{ki}\}$. For this linear combination a CLT is readily available. After tedious but straightforward computations we find that the process ζ^n has a weak limit which is a time-varying Brownian motion in $\mathbf{R}^K$ with covariance matrix $G(t) = [g_{ij}(t)]$ given by

$$g_{ij}(t) \quad = \begin{cases} \sum_k (1 + v_k^2 \mu_k^2 p_{ki} - p_{ki}) \mu_k p_{ki} c_k(x_t) + \mu_i^3 v_i^2 (1 - 2p_{ii}) c_i(x_t) & \text{if } i = j \\ \sum_k \mu_k c_k(x_t) p_{kj} p_{ki} (\mu_k^2 v_k^2 - 1) - \mu_j^3 v_j^2 c_j(x_t) - \mu_i^3 v_i^2 c_i(x_t) & \text{if } i \neq j \end{cases}$$

Proceeding as in Section 3 and using a second order Taylor expansion in the expression for $\eta^n(t)$ we can show that it has the same limit as $\lambda \int_0^t c'(x(s)) M(P - I) \xi^n(s) ds$, where $c'(x)$ is the gradient of c at x.

The above argument allows us to conclude that the weak limit of ξ^n is defined as the solution of the K-dimensional stochastic differential equation

$$d\xi(t) = \xi(t) F(t) dt + \sqrt{G(t)} dW(t) \, .$$

The notation used is: $F(t) = c'(x(t)) M(P - I)$, and $G(t)$ is the covariance matrix defined by (3). Note that $G(t)$ is nonnegative definite and has thus a well-defined square root.

5 Discussion

We started from a model of a "nonlinear" renewal process and showed how to obtain a functional law of large numbers and a functional central limit theorem. The methods used are martingale methods and a suitable decomposition for the central limit theorem. The advantage of these methods is that they enable one to foresee the form of the limiting process (i.e., the ordinary or stochastic differential equation in each case), and also that they give a unifying framework for proving convergence. It can be seen that the assumptions on the renewal character of the process can be considerably generalized. We chose a renewal process to make the presentation clear. In Section 3 we argued that the martingale central limit theorem is not directly applicable to the process ξ^n. We showed that a certain decomposition achieves to isolate the drift and diffusion parts and thus prove the result. The methods have been applied to a network of queues with state dependent service rates. For simplicity we treated the case of a closed network, although an open network can also be considered. It should be again mentioned that if the "heavy usage" condition is dropped then one should extend the results to cover the cases where the process is reflected at the boundaries. The methods to do this are extensions of the ones of Krichagina [11] and is the subject of further research.

References

[1] Anantharam, V. and Konstantopoulos, T. (1993). A functional central limit theorem for the jump processes of a uniform Markov process. Preprint, ECE Dept., U.T. Austin; submitted to *J. Appl. Prob.*

[2] Billingsley, P. (1968). *Convergence of Probability Measures.* J. Wiley.

[3] Brémaud, P. (1981). *Point Processes and Queues, Martingale Dynamics.* Springer, New York.

[4] Chen, H. and Mandelbaum, A. (1991). Discrete flow networks: bottleneck analysis and fluid approximations. *Math. Oper. Res.*, **16**, No. 2, 408-446.

[5] Chen, H. and Mandelbaum, A. (1991). Stochastic discrete flow networks: diffusion approximations and bottlenecks. *Ann. Prob.*, **19**, No.4, 1463-1519.

[6] Dacunha-Castelle, D. and Duflo, M. (1986). *Probability and Statistics.* Vol. 2. Springer, New York.

[7] Ethier, S. N. and Kurtz, T. G. (1986). *Markov Processes : Characterization and Convergence.* J. Wiley.

[8] Jacod, J. and Shiryayev, A. N. (1987). *Limit Theorems for Stochastic Processes.* Springer-Verlag, Berlin.

[9] Kogan, Ya. A. and Liptser, R. Sh. and Smorodinskii, A. V. (1986). Gaussian diffusion approximations of closed Markov models of computer networks. *Probl. Pered. Inform.*, **22**, No. 1, pp. 49-65 (English translation).

[10] Konstantopoulos, T., Papadakis, S.N., and Walrand, J. (1993). "Functional approximation theorems for controlled renewal processes". *J. Appl. Prob.*, to appear.

[11] Krichagina, E.V. (1992). Asymptotic analysis of queueing networks (martingale approach). *Stoch. Stoch. Rep.*, **40**, 43-76.

[12] Liptser, R. Sh. and Shiryayev, A. N. (1978). *Statistics of Random Processes II.* Springer-Verlag, Berlin.

[13] Liptser, R. Sh. and Shiryayev, A. N. (1980). A functional central limit theorem for semimartingales. *Th. Prob. Appl.*, **25**, No. 4, pp. 667-688.

[14] Liptser, R. Sh. and Shiryayev, A. N. (1989). *Theory of Martingales.* Kluwer, Boston.

Finite Moments for Inventory Processes

Karl Sigman* and David D. Yao†
Department of Industrial Engineering and Operations Research
Columbia University, New York, NY 10027-6699

Abstract

We study a continuous time inventory process that is a reflection mapping of a semimartingale *netput* process. Inventory processes of this type include the workload process in queues, dam and storage processes (with perhaps pure jump Levy input), as well as processes arising in fluid models. We establish sufficient conditions on the netput ensuring that the steady-state inventory has *finite* moments of order $k \geq 1$, and derive explicit bounds for these moments. The sufficient conditions require that the netput have a negative (local) drift, and that the (conditional) $(k+1)^{th}$ moment of its increments be bounded.

Keywords: Reflection mapping, finite moments, stationary ergodic, semimartingale, queues.

1 Introduction

The purpose of the present paper is to establish sufficient conditions that ensure finite moments of any desired order for the steady-state distribution of inventory processes. Let $Z(t)$ denote the amount of inventory at time t, satisfying the following *inventory equation*:

$$Z(t) = X(t) + L(t), \tag{1}$$

where $\{X(t) : t \geq 0\}, X(0) = 0$, is a given *netput* with negative drift (refer to the condition given in (2) below), and $L(t)$ is defined as

$$L(t) := \sup_{0 \leq s \leq t} X^-(s), \quad \text{with} \quad X^-(s) := -\min\{0, X(s)\}.$$

The setup here, in particular the term "netput", is borrowed from Harrison [9] (pp. 18-19), where X is a Brownian motion. The mapping in (1), which takes a path of X and maps it to a path of Z, is sometimes called the *reflection mapping* and is of fundamental importance in a wide variety of applications both deterministic and stochastic. See, for example, Borovkov [3] (Chapter 1, Section 6), Chung and Williams [4] (Section 8.2), El Karoui and Chaleyat-Maurel [6], Glynn [8], and Prabhu [13], in addition to [9] cited above. Recent applications to inventory processes also appeared in Bardhan and Sigman [1], [2].

---

*Research supported in part by NSF grant DDM-89-57825.
†Research supported in part by NSF grant DDM-91-08540

We study two cases: (a) the netput X has a bounded variation, and (b) X is a semimartingale (martingale plus a process with bounded variation). Although (a) is a special case of (b), we choose to focus on (a) first so as to bring out the basic ingredients of our approach, which then generalizes easily to case (b) through incorporating certain martingale inequalities.

In the case of bounded variation, $X(t)$ is of the form $A(t) - B(t)$, where $A = \{A(t) : t \geq 0\}$ and $B = \{B(t) : t \geq 0\}$ are two non-negative and non-decreasing processes representing the input and the *potential* output, respectively. For example, in a standard single-server queueing model with $Z(t)$ denoting total workload, we have $B(t) = t$ and $A(t) = \sum_{n=1}^{N(t)} S_n$, where $\{N(t)\}$ denotes the counting process for arrivals, and $\{S_n\}$ denotes customer service times. In this case, $L(t)$ is precisely the total idle time of the server during $[0, t]$. On the other hand, in an inventory model the input $A(t)$ need not be generated by a point process of arrivals; it can be a continuous flow process (e.g. have continuously differentiable sample paths), or a pure jump process (such as a pure jump Levy process with infinite Levy measure).

We derive sufficient conditions (not necessary in general) ensuring that the steady-state inventory has finite k^{th} moments, $k \geq 1$, and obtain explicit bounds for these moments. The sufficient conditions require that the netput have a negative local drift, and that the conditional $(k+1)^{th}$ moments of its increments be bounded. In applications, finite moments are important performance measures. For instance, to guarantee the quality of service in communication systems, it is usually required that the expected message delay be bounded by a given constant. Hence, our conditions for finite moments and the computable bounds provide practical means to support system design without having to make unnecessary independence and distributional assumptions on the underlying processes.

Our results are motivated by the classical finite moment conditions of Kiefer and Wolfowitz [11] for queues with renewal input and i.i.d. services, as well as the more recent works of Wolff [14] and Daley and Rolski [5]. By adopting the setup in (1), our results, even in the bounded variation case, are considerably more general than those for queueing models. Our approach, on the other hand, is largely influenced by the recent work of Meyn and Down [12], which studied the first moment of delay in a generalized Jackson network.

In Section 2 we present the results for the case of bounded variation, followed by several examples in Section 3. Section 4 extends the results to the semimartingale case. Due to page limit, we have removed all proofs, which are detailed in a full paper, available upon request.

2 The Case of Bounded Variation

All stochastic processes below are assumed to have sample paths that are right continuous with left limits (e.g., in the space $\mathcal{D}[0,\infty)$ endowed with the Skorohod topology; see, e.g., Ethier and Kurtz [7]). Let the netput $X = \{X(t) : t \geq 0\}$, $X(0) = 0$, be a stochastic process on an underlying probability space, $(\Omega, \mathcal{F}, P)$, with stationary and ergodic increments, and *negative drift*:

$$\lim_{t\to\infty} X(t)/t = \mathsf{E}[X(1)] = -\alpha < 0, \quad \text{a.s.,} \tag{2}$$

where $\alpha > 0$ is a constant. (Note the negative drift is a *stability* condition.) The inventory process Z is then constructed via (1) with $Z(0) = 0$. This ensures the existence of a unique proper steady-state distribution for Z, as well as the ergodicity of Z; refer to [3] (Chapter 1, Section 6) and [1] for details. Hence, in particular, for any non-negative Borel measurable function $g : \Re_+ \mapsto \Re_+$, we have

$$\lim_{t\to\infty} \frac{1}{t}\int_0^t g(Z(s))ds = \mathsf{E}[g(\hat{Z})], \quad \text{a.s.,} \tag{3}$$

where $\hat{Z}$ denotes a random variable having the steady-state distribution:

$$\mathsf{P}(\hat{Z} \leq x) = \lim_{t\to\infty} \mathsf{P}(Z(t) \leq x). \tag{4}$$

Let $\mathcal{F}_t = \sigma\{X(s) : 0 \leq s \leq t\}$ be our filtration. From (1), clearly $Z(t) \in \mathcal{F}_t$ for all t. Denote $\mathsf{E}_{\mathcal{F}_t}[X] := \mathsf{E}[X|\mathcal{F}_t]$. Assume that the netput is of the form $X(t) = A(t) - B(t)$, where A and B are both non-negative and non-decreasing processes, $A(0) = 0, B(0) = 0$, with jointly stationary ergodic increments. Let

$$A(t,h):=A(t+h) - A(t) \quad \text{and} \quad B(t,h):=B(t+h) - B(t)$$

denote the increments; and similarly, $X(t,h) := X(t+h) - X(t)$.

The key conditions for our main result (Theorem 1 below) are as follows:

A1: There exist constants $\hat{\alpha} > 0$ and $\ell \geq 0$ such that for all $t \geq 0$ and $h > 0$,

$$\mathsf{E}_{\mathcal{F}_t}[X(t,h)] \leq -\hat{\alpha}h + \ell.$$

A2(k) ($k \geq 1$ is an integer): There exist functions $a_k : \Re_+ \mapsto \Re_+$ and $b_k : \Re_+ \mapsto \Re_+$, such that for all $t \geq 0$ and $h > 0$, $\int_0^h a_k(s)ds < \infty$,

$$\mathsf{E}_{\mathcal{F}_t}[A^{k+1}(t,h)] \leq a_k(h), \quad \text{and} \quad \mathsf{E}_{\mathcal{F}_t}[B^{k+1}(t,h)] \leq b_k(h).$$

Theorem 1 Suppose for a given $k \geq 1$, the netput $X = A - B$ satisfies **A1** and **A2(k)**. Then, there exists a constant $c_k < \infty$, computable in terms of a_k and b_k, such that

$$\overline{\lim_{t\to\infty}} \frac{1}{t} \int_0^t \mathsf{E}[Z^k(s)]ds \leq c_k, \tag{5}$$

and

$$\lim_{t\to\infty} \frac{1}{t} \int_0^t Z^k(s)ds = \mathsf{E}[\hat{Z}^k] \leq c_k, \quad \text{a.s.}, \tag{6}$$

where $\hat{Z}$ follows the steady-state distribution of Z in (4).

Note that **A1** is a *local* drift condition, ensuring that the input does not overpower the output in small periods of time, and **A2(k)** ensures that the $(k+1)^{th}$ conditional moment of the input and the output in small intervals is not too large. It is easily seen that under **A1** it is necessary that $\alpha \geq \hat{\alpha}$, since

$$\mathsf{E}[X(t,h)] = \mathsf{E}[X(h)] = -\alpha h$$

follows from the stationary increments and (2), while **A1** implies (via taking expectation on both sides of the inequality)

$$\mathsf{E}[X(t,h)] \leq -\hat{\alpha}h + \ell.$$

Hence, $-\alpha h \leq -\hat{\alpha}h + \ell$ for all h, implying $\alpha \geq \hat{\alpha} > 0$. (In other words, **A1** implies the negative drift condition in (2).)

To establish Theorem 1, we need several lemmas. The first one, an elementary result, is in the same spirit as (but simpler than) Lemma 2.3 of [12].

Lemma 1 Let f, g and c be finite functions on $\Re_+$, with f and g non-negative and Lebesgue measurable. If for some fixed $h > 0$

$$f(s+h) \leq f(s) - g(s) + c(h), \quad \forall s \in \Re_+,$$

then

$$\frac{1}{t}\int_0^t g(s)ds \leq c(h) + \frac{1}{t}\int_0^h f(s)ds, \quad \forall t > 0.$$

To present the next two lemmas, we need more notation. From (1), it follows that Z satisfies the recursion

$$Z(t+h) = Z(t) + X(t,h) + L(t,h), \tag{7}$$

where $L(t,h) := L(t+h) - L(t)$, and it is known ([4], p. 149) that

$$L(t,h) = \sup_{0\leq u\leq h} [Z(t) + X(t,u)]^-. \tag{8}$$

In addition, define

$$D(t) := A(t) - Z(t) = B(t) - L(t), \quad \text{and} \quad D(t,h) := B(t,h) - L(t,h).$$

Intuitively, $D = \{D(t) : t \geq 0\}$ is the *throughput* process: $D(t)$ is the amount of input that has been processed during $[0,t]$; and $D(t,h)$ denotes the amount of input that is processed during $[t, t+h]$. Finally, let

$$\Delta(t,h) := A(t,h) - D(t,h) = X(t,h) + L(t,h),$$

and rewrite (7) as

$$Z(t+h) = Z(t) + \Delta(t,h). \tag{9}$$

It is useful to keep in mind the following facts: $A(t)$ and $B(t)$ are both non-decreasing by definition, and so is $L(t)$. Hence, $A(t,h)$, $B(t,h)$ and $L(t,h)$ are all non-negative and non-decreasing in h.

Lemma 2 For any non-negative t and h,
(i) $L(t,h) \leq B(t,h)$, and hence $D(t,h) \geq 0$;
(ii) $Z(t)L(t,h) \leq B^2(t,h)$.

Remark 1 A well-known property of the reflection mapping is that $L(t)$ remains a constant when $Z(t) > 0$. (See, e.g., [9], p. 20.) In other words, $Z(t) > 0$ implies $L(t,\epsilon) = 0$ for a sufficiently small $\epsilon > 0$. This property is now strengthened by the fact, brought up in the proof of Lemma 2(ii), that $Z(t) > B(t,h) \geq 0$ implies $L(t,h) = 0$ for *any* positive h.

Lemma 3 (i) Suppose **A1** and **A2**(1) hold. Then, for any $t \geq 0$ and $h > 0$,

$$Z(t)\mathsf{E}_{\mathcal{F}_t}[\Delta(t,h)] \leq -(\hat{\alpha}h - \ell)Z(t) + b_1(h).$$

(ii) Suppose **A2**(j) holds for some $j \geq 1$. Then, for any $t \geq 0$ and $h > 0$,

$$\mathsf{E}_{\mathcal{F}_t}[|\Delta(t,h)|^{j+1}] \leq a_j(h) + b_j(h).$$

Corollary 1 Under the conditions of Theorem 1, the inequalities in (5) and (6) hold for all $j = 1, \ldots, k$.

Remark 2 Suppose $B(t,h) \geq H(t,h)$ for all non-negative t and h, where H is a non-negative and nondecreasing process. Then, if A and H (rather than B) satisfy the conditions of Theorem 1, then the conclusions there still hold. This follows since $Z(t) \leq Y(t)$ for all t where Y is the inventory process corresponding to the netput $A - H$. This illustrates the fact that the moment condition on B is certainly not a necessary one.

Remark 3 Our assumption that $Z(0) = 0$ is not needed; all the results hold for $Z(0) \geq 0$ arbitrary as long as $\mathsf{E}[Z^{k+1}(0)] < \infty$. We must, however, include $Z(0)$ into the filtration $\mathcal{F}_t = \sigma\{X(u) : 0 \leq u \leq t\}$, and verify that **A1** and **A2**(k) still hold. The most convenient setup is to allow an arbitrary $X(0) \geq 0$ and define Z using (1) exactly as before; then $Z(0) = X(0)$. A special case is when $Z(0)$ is independent of X and $\mathsf{E}[Z^{k+1}(0)] < \infty$.

Remark 4 If the increments of X are stationary but not ergodic, then Theorem 1 still holds, except that (6) must be changed to $\mathsf{E}[\hat{Z}^k] \leq c_k$. Letting $\mathcal{I}$ denote the invariant σ-field for the increments of X, we have

$$\lim_{t\to\infty} \frac{1}{t} \int_0^t Z^k(s)ds = \mathsf{E}_{\mathcal{I}}[\hat{Z}^k] < \infty, \quad \text{a.s.}$$

However, $\mathsf{E}_{\mathcal{I}}[\hat{Z}^k]$ is a random variable, which in general is not bounded by c_k. Furthermore, our framework only requires that X be *asymptotically stationary*, which means that for $X_s{:=}\{X(s+t) - X(s) : t \geq 0\}$ denoting the shifted increments (by time s), the distributions $P(X_s \in \cdot)$ on the Borel sets of $\mathcal{D}[0,\infty)$ must converge in total variation in mean, as $s \to \infty$, to a probability measure (see [1] for details). This includes a wide variety of netput such as those with increments having some kind of positive recurrent regenerative structure. Finally, observe that to obtain (5), even an asymptotically stationary X is not required; only **A1** and **A2**(k) are needed.

Remark 5 In the proof of Theorem 1 (not given here) we have in fact also found the following simple algorithm to compute the bounds $\{c_k\}$ in (5) and (6). Suppose **A1** and **A2**(n) hold. Then, $a_n(h)$, $b_n(h)$, $\hat{\alpha}$ and ℓ are known. For $j = 1, \ldots, n-1$, set

$$a_j(h) := [a_n(h)]^{j/n}, \quad b_j(h) := [b_n(h)]^{j/n}.$$

Since the first moment is perhaps the most important among all moments, pick h so as to minimize c_1:

$$\min_{h > \ell/\hat{\alpha}} c_1 := \frac{a_1(h) + 3b_1(h)}{2(\hat{\alpha}h - \ell)}.$$

By definition, $c_0 := 1$. For $k = 2, \ldots, n$, use the following recursion:

$$c_k = (\hat{\alpha}h - \ell)^{-1} \left[\frac{1}{k+1} \sum_{j=0}^{k-1} \binom{k+1}{j} c_j[a_{k-j}(h) + b_{k-j}(h)] + c_{k-1}b_1(h) \right].$$

Of course, the choice of h can address other objectives as well.

3 Examples

Example 1 *Workload in a standard single-server queue.* In this case $B(t) = t$, i.e., the server processes work (if available) at a constant, unit rate. Let $\{(t_n, S_n) : n \geq 1\}$ be a stationary ergodic marked point process of customer arrival times t_n, and service times S_n. Let $N(t)$ denote the counting process for arrivals. Then,

$$A(t) = \sum_{n=1}^{N(t)} S_j$$

denotes the amount of work arrived in $(0, t]$. The workload process, $Z(t)$, follows (1): $Z(t) = A(t) - t + L(t)$. Note that $B(t) = t$ satisfies **A2**(k) for any $k \geq 1$, so if we want to apply Theorem 1 for the k^{th} moment of the workload, we need constants $\hat{\alpha} > 0$ and $\ell \geq 0$ such that for all $t, h \geq 0$,

$$\mathsf{E}_{\mathcal{F}_t}[A(t,h) - h] \leq -\hat{\alpha}h + \ell,$$

together with the existence of a function $a_k : \Re_+ \mapsto \Re_+$ such that for all $h \geq 0$,

$$\mathsf{E}_{\mathcal{F}_t}[A^{k+1}(t,h)] \leq a_k(h).$$

As a special case, if service times are i.i.d. and independent of the arrival times $\{t_n\}$, then the above conditions will hold if

$$\mathsf{E}[S_n^{k+1}] < \infty,$$

and if there exist constants $\gamma > 0$ and $\ell \geq 0$ such that

$$\mathsf{E}_{\mathcal{F}_t}[N(t,h)] \leq \gamma h + \ell,$$

with

$$\gamma^{-1} > E(S_n).$$

When $k = 1$, these are exactly the conditions required in Theorem 2.1 of [12].

Example 2 *Single-server queue with varying service rate.* As in Example 1, Z denotes the workload process in a system with $A(t) = \sum_{n=1}^{N(t)} S_j$. However, here the server serves at rate $r(t)$ at time t, where jointly with A, $\{r(t) : t \geq 0\}$ is a stationary ergodic process, and A has stationary ergodic increments. Here, $B(t) = \int_0^t r(s)ds$, and $B(t,h) = \int_t^{t+h} r(s)ds$. If for all $t \geq 0$, $r(t)$ is bounded almost surely: $m \leq r(t) \leq M$ for some constants $0 < m < M < \infty$, then

$$\mathsf{E}_{\mathcal{F}_t}[B(t,h)] \geq mh, \quad \text{and} \quad \mathsf{E}_{\mathcal{F}_t}[B^k(t,h)] \leq (Mh)^k.$$

If additionally, A satisfies **A2**(k) and also

$$\mathsf{E}_{\mathcal{F}_t}[A(t,h)] \leq \hat{\rho}h, \quad \text{with} \quad \hat{\rho} - m < 0,$$

then both **A1** and **A2**(k) are satisfied.

Example 3 *Pure jump Levy input.* $B(t) = t$, and $A(t)$ is a pure jump Levy process with infinite Levy measure (the number of jumps in any finite interval is infinite, see [13], Section 3.2). Levy processes have independent as well as stationary increments. In particular, consider the case when $A(t) \sim \Gamma(\lambda, \gamma t)$. Then

$$\mathsf{E}[e^{iuA(t)}] = \left(\frac{\lambda}{\lambda - iu}\right)^{\gamma t},$$

from which we obtain $\mathsf{E}[A(t)] = \gamma t/\lambda$ and $\mathsf{E}[A^2(t)] = \gamma t(\gamma t + 1)/\lambda^2$. In this case $\mathsf{E}[X(t,h)] = \gamma h/\lambda - h$, and hence $\alpha > 0$ iff $\lambda > \gamma$. By independent increments, we also have **A2**(1) satisfied with $a_1(t) = \mathsf{E}[A^2(t)] = \gamma t(\gamma t + 1)/\lambda^2$ and $b_1(t) = t^2$. Also by independent increments, **A1** is satisfied with $\hat{\alpha} = \alpha$, and thus we see that whenever $\lambda > \gamma$, Theorem 1 applies and $\mathsf{E}[\hat{Z}] < \infty$. The finiteness of higher moments is similarly argued. (This example is meant for illustrative purpose only; even the more general case, with the netput X being an arbitrary Levy process with negative drift, is classic and in fact already well understood; see, e.g., [13], and [1], Section 4.)

Finally, we remark that based on an obvious discrete-time analogy of Theorem 1, we can also reproduce the known finite-moment results for customer delays in queues, e.g., those in [11] and [14].

4 The Semimartingale Case

Here we assume that the netput process X is a semimartingale; specifically, X can be expressed as

$$X(t) = M(t) + A(t) - B(t),$$

where $\{M(t)\}$ is a martingale with respect to the filtration $\{\mathcal{F}_t\}$, and $\{A(t)\}$ and $\{B(t)\}$ are two non-decreasing processes as before. Assume all processes have stationary increments and paths in $\mathcal{D}[0,\infty)$. It is known that this model covers a wide range of processes.

Let $M(t,h) := M(t+h) - M(t)$. Other notations are same as before. Suppose **A1** is still in force, while **A2** is strengthened as follows:

A2'(k) ($k \geq 1$ is an integer): In addition to **A2**(k), there also exists a function $m_k : \Re_+ \mapsto \Re_+$, such that for all $t \geq 0$ and $h > 0$, $\int_0^h m_k(s)ds < \infty$, and

$$\mathsf{E}_{\mathcal{F}_t}[|M(t,h)|^{k+1}] \leq m_k(h).$$

(The above condition is easily verified for Brownian motion, and is easy to check for more general Levy processes.)

Recall the two results in Lemma 3 are key to the proof of Theorem 1. Both rely crucially on Lemma 2. So here we start with modifying Lemma 2.

Lemma 4 For a semimartingale netput process X, and any non-negative t and h, we have
(i) $L(t,h) \le \sup_{0\le u\le h} |M(t,u)| + B(t,h)$; and
(ii) $Z(t)L(t,h) \le [\sup_{0\le u\le h} |M(t,u)| + B(t,h)]^2$.

To generalize Lemma 3 based on Lemma 4, we need the following adaptation of Doob's L_p inequality:

Lemma 5 Let $\{Y(u),\ u \ge 0\}$ be a non-negative submartingale with respect to a filtration $\{\mathcal{F}_u\}$. Then, for $p > 1$, and $h > 0$,

$$\mathsf{E}_{\mathcal{F}_0}[\sup_{0\le u\le h} Y^p(u)] \le \left(\frac{p}{p-1}\right)^p \mathsf{E}_{\mathcal{F}_0}[Y^p(h)] \quad \text{a.s.}$$

Note that replacing $\mathsf{E}_{\mathcal{F}_0}$ by E in the above inequality recovers the original Doob's L_p inequality (see, e.g., Karatsas and Shreve [10], p. 14). On the other hand, for any $\mathbf{A} \in \mathcal{F}_0$, since $\{Y(u)\mathbf{1}(\mathbf{A})\}$ is still a non-negative submartingale, applying Doob's L_p inequality, we have

$$\mathsf{E}[\sup_{0\le u\le h} Y^p(u)\mathbf{1}(\mathbf{A})] \le \left(\frac{p}{p-1}\right)^p \mathsf{E}[Y^p(h)\mathbf{1}(\mathbf{A})]$$

(noticing that $[\mathbf{1}(\mathbf{A})]^p = \mathbf{1}(\mathbf{A})$), hence the desired a.s. inequality in Lemma 5.

Lemma 6 (i) Suppose **A1** and **A2'**(1) hold. Then, for any $t \ge 0$ and $h > 0$,

$$Z(t)\mathsf{E}_{\mathcal{F}_t}[\Delta(t,h)] \le -[\hat{\alpha}h - \ell]Z(t) + \left[2m_1^{1/2}(h) + b_1^{1/2}(h)\right]^2.$$

(ii) Suppose **A2'**(j) holds for some $j \ge 1$. Then, for any $t \ge 0$ and $h > 0$,

$$\begin{aligned}\mathsf{E}_{\mathcal{F}_t}[|\Delta(t,h)|^{j+1}] \ &\le \ \left[(2+1/j)m_j^{1/(j+1)}(h) + a_j^{1/(j+1)}(h)\right]^{j+1} \\ &+ \ \left[m_j^{1/(j+1)}(h) + b_j^{1/(j+1)}(h)\right]^{j+1}.\end{aligned}$$

With the above lemma, we can establish

Theorem 2 When the netput process X is a semimartingale, Theorem 1 and Corollary 1 still hold, with **A2**(k) strengthened to **A2'**(k).

References

[1] I. Barhdhan and K. Sigman. Stationary Regimes for Inventory Processes. Technical Report, (1992).

[2] I. Barhdhan and K. Sigman. Rate Conservation Law for Stationary Semimartingales. *Probability in the Engineering and Informational Sciences*, **7** (1993), 1-17.

[3] A.A. Borovkov, *Stochastic Processes In Queueing Theory.* Springer-Verlag (1976).

[4] K.L. Chung and R.J. Williams, *Introduction to Stochastic Integration.* Birkhauser (1983).

[5] D. Daley and T. Rolski. Finiteness of waiting-time moments in general stationary single-server queues. *Annals of Applied Probability,* **2** (1992), 987-1008.

[6] N. El Karoui and M. Chaleyat-Maurel. Un Probleme de reflexion et ses applications au temps local et aux equations differentielles stochastiques sur $\Re$ — Cas continu. *Asterisque, Societe Mathematique de France*, **52-53** (1978), 117-144.

[7] S. Ethier and T. Kurtz, *Markov Processes, Characterization and Convergence.* John Wiley and Sons (1986).

[8] P. Glynn. Diffusion Approximations. *Handbooks on OR & MS*, Ch. 4, D.P. Heyman and M.J. Sobel (eds.). Elsevier Science Publishers (1990).

[9] J. M. Harrison, *Brownian Motion and Stochastic Flow Systems.* John Wiley and Sons (1985).

[10] I. Karatsas and S.E. Shreve. *Brownian Motion and Stochastic Calculus.* Springer-Verlag (1988).

[11] J. Kiefer and J. Wolfowitz. On the characteristics of the general queueing process with applications to random walk. *Ann. Math. Statist.*, **27** (1956), 147-161.

[12] S.P. Meyn and D. Down. Stability of Generalized Jackson Networks. *The Annals of Applied Probability* (to appear).

[13] N.U. Prabhu. *Stochastic Storage Processes.* Prentice-Hall (1980).

[14] R. W. Wolff. On finite delay moments in queues. *Operations Research*, **39** (1991), 771-775.

Optimal control of manufacturing systems with production dependent machine failures

Guo-Xiang Yu and Pirooz Vakili *
Department of Manufacturing Engineering
Boston University

Sept. 1993

Abstract

In this paper we study the optimal control of manufacturing systems with random machine failures. The machine failure and repair is modeled as a Markov chain and the failure rate is dependent on the production rate. We first explore the optimal control within the hedging point policies. Then we derive an analytical approach to evaluate the value function under piecewise constant control policies. The nonlinear optimization procedure is proposed to investigate the optimal control policies. Three examples are given to show that the optimal control may depend on the initial state and the structure of the failure rate function.

1 Introduction

Using fluid model to study the manufacturing systems with random machine failures has been a recent theoretical approach among the researchers. Kimemia and Gershwin [5] described a general multi-machine and multi-part-type model and suggested that the optimal control policy is Hedging Point Policy (HPP) (or threshold policy). Akella and Kumar [1] solved the optimal control thoroughly for a simple one machine one part-type system with discounted cost. The optimal control is found to be a HPP. Later Bielecki and Kumar [2] solved the same problem with average cost and the optimal control found is also HPP. Sharifnia [7] showed how to calculate the average cost under HPP and locate the optimal hedging points. Later Hu and Xiang [4] found the equivalence of the fluid model to the queueing systems. All the above work assumed that the machine failure is independent of the production, or the failure rate is a constant. In reality, however, this assumption is often violated and the machine failure usually depends on many factors, especially the speed of machine operation. In [3], Hu et.al. studied the single machine system assuming that the failure rate is dependent on the production rate. It is found

---

*The work in this paper was supported in part by the National Science Foundation under Grants EID-9212122.

that the HPP is optimal if and only if the failure rate function is affine. Liberopoulos and Caramanis [6] used probability density to calculate the average cost under HPP and explored the optimal HPP numerically for different failure functions.

In this paper we shall continue the work done in [3] and explore the optimal control policy for general failure rate functions. The control policy assumed in [3] is the feedback control which only depend on the surplus state. That means if a control policy is optimal then it is optimal no matter what is the initial state started. In this paper, we shall relax the control policy such that it may depend on the initial state. Given that relaxation, an optimal policy will be dependent not only on the surplus, but also on the initial state. We shall explore in this paper the optimal HPP as well as the optimal control policy for the single machine system under the relaxed optimality.

The rest of the paper is organized as follows. In Section 2 we describe the system model and some preliminary results. In Section 3, we explore the optimal HPP among all HPP's. In section 4, we solve analytically the value function under the piecewise constant controls and propose an nonlinear optimization procedure to find the optimal piecewise constant control policy which is a good approximation to the real optimal policy. Three examples is investigated in Section 5 for linear, convex and concave failure functions. We conclude in Section 6.

2 Preliminaries

The manufacturing system under consideration has a single production machine and produces a single part-type. There is a constant demand with rate d that the system must satisfy and the backlog is allowed. The machine is subject to random failures and repairs. Let $\alpha \in \{1,2\}$ denote the machine state where 1 correspond to machine is operational and 2 to down. The transition of the machine state follows a Markov chain with repair rate q_2 and failure rate $q_1(u)$ where u is the instantaneous production rate. When machine is operational it can produce with rate between 0 and $\bar{u}(> d)$. Let x denote the surplus of the product. We assume that x changes continuously, so that the dynamics of the surplus x is governed by differential equation:

$$\frac{dx(t)}{dt} = u(t) - d \tag{1}$$

In order for the system to meet the demand in the long run the following stability condition must be satisfied: $q_2(\bar{u} - d) - q_1(\bar{u})d > 0$. Throughout this paper we assume this stability condition holds. There is a unit cost of c^+ for holding inventory and a unit cost of c^- for backlog. The objective is to minimize the following long-term discounted cost (value function):

$$W_i(x) = E\left[\int_0^\infty e^{-\gamma t}(c^+x^+(t) + c^-x^-(t))dt \mid \alpha(0) = i, x(0) = x\right] \tag{2}$$

where γ is the discount factor and $x^+ = \max\{0, x\}$, $x^- = \max\{0, -x\}$.

The HPP is defined via

$$u = \begin{cases} 0 & \text{if } x > z \\ d & \text{if } x = z \\ \bar{u} & \text{if } x < z \end{cases} \tag{3}$$

where z is called the hedging point. Let $W_i^z(x)$ be the value function associated with the above HPP. If the HPP is used, not surprisingly, the value functions only depend on the values of the failure rate function at $\bar{u}$, d and 0; we denote these values by $q_{12} = q_1(\bar{u})$, $q_{11} = q_1(d)$, and $q_{10} = q_1(0)$. Also let

$$A_1 = \begin{pmatrix} \frac{\gamma+q_{12}}{\bar{u}-d} & -\frac{q_{12}}{\bar{u}-d} \\ \frac{q_2}{d} & -\frac{\gamma+q_2}{d} \end{pmatrix}, \quad A_2 = \begin{pmatrix} -\frac{\gamma+q_{10}}{d} & \frac{q_{10}}{d} \\ \frac{q_2}{d} & -\frac{\gamma+q_2}{d} \end{pmatrix}, \quad b_1 = \begin{pmatrix} -\frac{1}{\bar{u}-d} \\ \frac{1}{d} \end{pmatrix}, \quad b_2 = \begin{pmatrix} \frac{1}{d} \\ \frac{1}{d} \end{pmatrix}.$$

The value function $W^z(x) = (W_1^z(x), W_2^z(x))^\tau$ can be solved analytically and the solution for $z \geq 0$ is (see [3])

$$W^z(x) = \begin{cases} e^{A_1 x}[W^z(0) - A_1^{-2} b_1 c^-] + A_1^{-1} b_1 c^- x + A_1^{-2} b_1 c^- & x \leq 0 \\ e^{A_1 x}[W^z(0) + A_1^{-2} b_1 c^+] - A_1^{-1} b_1 c^+ x - A_1^{-2} b_1 c^+ & 0 < x < z \\ e^{A_2(x-z)}[W^z(z) + A_2^{-1} b_2 c^+ z + A_2^{-2} b_2 c^+] & \\ -A_2^{-1} b_2 c^+ x - A_2^{-2} b_2 c^+ & x \geq z \end{cases} \tag{4}$$

where

$$W^z(z) = \begin{pmatrix} 1 \\ 1 + \frac{\gamma}{q_{11}} \end{pmatrix} W_1^z(z) - \begin{pmatrix} 0 \\ \frac{c^+}{q_{11}} \end{pmatrix} z \tag{5}$$

$$W_1^z(z) = \frac{c^+ z}{\gamma} + \frac{q_{11}[(c^+ + c^-)e^{\lambda_- z} - c^+][q_{12} d - (\bar{u} - d)(\lambda_- d + \gamma + q_2)]}{\gamma^2[q_{12} q_{11} - q_{12}\lambda_- d + q_{11}(\lambda_- d + \gamma + q_2)]} \tag{6}$$

(λ_- is the only negative eigenvalue of A_1). The value of $W^z(0)$ can be obtained by the continuity at $x = 0$.

From the results in [3] we know that the hedging point policy is optimal if and only if the failure function is affine. The definition of the optimality in [3] implicitly assume that an optimal control policy is independent of the initial states. That is, if policy π^* is optimal then

$$W_i^{\pi^*}(x) = \inf_{\pi} W_i^{\pi}(x) \qquad \text{for all } i \text{ and } x$$

In this paper, we shall relax the definition of optimality and allow an optimal control to depend on the initial state. In other words, a control policy $\pi^*(i, x)$ is optimal if $W_i^{\pi^*(i,x)}(x)$ is minimized among all admissible policies for the fixed initial state (i, x). In the following section, we first explore the optimal HPP among all HPP's.

3 Optimal hedging point policy

To simplify the discussions we assume that $q_{10} = 0$ and $q_{12} > q_{11}$. These are quite reasonable assumptions noticing that the former means the machine will never fail if it is idle and the latter means that the machine is more likely to fail when producing at the maximal speed than at demand speed.

It can be proved that a negative hedging point can never be optimal (see [8]). Hence it suffices to consider all nonnegative hedging points. From Equation (4), it follows that $W^z(x)$ is differentiable w.r.t. z at all points except $z = x$. Hence, to find the optimal z we need to locate the zero of $\frac{d}{dz}W^z(x)$ or explore the monotonicity of $W^z(x)$.

It can be calculated that the derivative of the value function is:

$$\frac{d}{dz}W^z(x) = \begin{cases} H_x(z), & \text{if } x < z \\ G_x(z), & \text{if } x > z \end{cases}$$

where

$$H_x(z) = e^{A_1(x-z)}\left[\frac{d}{dz}W^z(z) - A_1W^z(z) - b_1c^+z\right],$$

$$G_x(z) = e^{A_2(x-z)}\left[\frac{d}{dz}W^z(z) - A_2W^z(z) - b_2c^+z\right].$$

We have the following results for the function $H_x(z)$ and $G_x(z)$. The proof can be found in [8].

Lemma 1 *$H_x(z)$ either is always positive or has a unique zero at*

$$z_1 = \frac{1}{\lambda_-}\log\left[\frac{c^+\gamma(\gamma + d\lambda_- + q_2)[(\gamma + q_2)(\bar{u} - d) + \bar{u}q_{11} - dq_{12}]}{(c^+ + c^-)[\gamma(\gamma + q_{11} + q_2) + d\lambda_-(\gamma + q_{11})][(\gamma + q_2 + d\lambda_-)(\bar{u} - d) - dq_{12}]}\right] \tag{7}$$

In the latter case, $H_x(z) < 0$ when $z < z_1$ and $H_x(z) > 0$ when $z > z_1$.

Lemma 2 *Let $G_x^i(z)$ denotes the i-th entry of $G_x(z)$. Then $G_x^1(z)$ has a unique zero at*

$$z_2 = \frac{1}{\lambda_-}\log\left[\frac{c^+\gamma[q_{11}\bar{u}(\gamma + q_2 + d\lambda_-) - d^2\lambda_- q_{12}]}{(c^+ + c^-)q_{11}(\gamma + d\lambda_-)[(\gamma + q_2 + d\lambda_-)(\bar{u} - d) - dq_{12}]}\right], \tag{8}$$

and $G_x^1(z) < 0$ when $z < z_2$ and $G_x^1(z) > 0$ when $z > z_2$. If $z > \max\{z_1, z_2\}$, then $G_x^2(x) > 0$, and if $z < \min\{z_1, z_2\}$, then $G_x^2(z) < 0$.

By these two lemmas, it is not difficult to locate the optimal hedging points for different initial states. To simplify discussions, we assume $z_1 = 0$ if $H_x(z)$ is always positive. We summarize the results in Table 3 and 1 for the case that both z_1 and z_2 are positive, and in Table 4 and 2 for the case that one of z_1 and z_2 is positive. If both z_1 and z_2 are negative we conclude that zero is the optimal hedging point.

Now we explore the order of the two points z_1 and z_2. If at least one of the points is negative or does not exist, the problem becomes trivial and unnecessary. We assume z_1 and z_2 are both positive in the rest of this section. It is not difficult to obtain that $e^{\lambda_- z_1} - e^{\lambda_- z_2} = k(\bar{u}q_{11} - dq_{12})$ where

$$k = \frac{c^+d\gamma^2\left[\gamma(\gamma + q_{12} + q_2) + d\lambda_-(\gamma + q_{12})\right]}{(c^+ + c^-)(\bar{u} - d)q_{11}(\gamma + d\lambda_-)\left[q_{12}d - (\bar{u} - d)(\gamma + q_2 + d\lambda_-)\right]\left[\gamma(\gamma + q_2 + q_{11}) + d\lambda_-(\gamma + q_1\right.}$$

It immediately follows that

Lemma 3 *The two hedging points z_1 and z_2, given by Equation (7) and (8) respectively, are equal if and only if the linearity condition $q_1(d)\bar{u} = q_1(\bar{u})d$ holds.*

A special case for the failure function is $q_1(u) = au^\beta$. It is clear that in this case we have

$$dq_{12} - \bar{u}q_{11} \begin{cases} < 0; & \text{if } \beta < 1 \\ > 0; & \text{if } \beta > 1 \end{cases}$$

It can be proved that the sign of k is absolutely decided by the sign of $\gamma(\gamma + q_2 + q_{11}) + d\lambda_-(\gamma + q_{11})$. We have the following results (see [8] for proof)

Lemma 4 *If $\beta < 1$ then $z_1 > z_2$. If $\beta > 1$ then $z_1 < z_2$ if $\gamma(\gamma+q_{11}+q_2)+d\lambda_-(\gamma+q_{11}) < 0$ and $z_1 > z_2$ if $(dq_{12} - \bar{u}q_{11})(\gamma + q_{12}) - q_2q_{12}(\bar{u} - d) > 0$.*

From these results we see that the optimal hedging point is not unique for non-affine failure functions. If we do not allow the control policy depend on the initial state, as in [3], then there does not exist an optimal hedging point. This result gives us an explanation to why the linearity condition is necessary and sufficient for the optimality of the hedging point policy obtained in [3].

4 Value functions under piecewise constant control policies

In this section we shall derive the analytic solutions to the value function under piecewise constant control policy. The piecewise constant control policy we used is briefly described as follows. First we divide the state space R into many regions. In each region the production rate is a constant. Except for upper and lower region which the control is fixed to 0 and $\bar{u}$ respectively, in each region the control can be chosen arbitrarily subjected to capacity constraint.

Let $P_l < 0$ and $P_u > 0$ are two real number and n_l and n_u are two positive integers. Now divide $[P_l, 0]$ and $[0, P_u]$ equally into n_l and n_u sub-regions respectively, and denote

$$z_0 = P_l; \quad z_i = z_{i-1} - \frac{P_l}{n_l}, \ i = 1, \ldots, n_l; \quad z_j = z_{j-1} + \frac{P_u}{n_u}, \ j = n_l + 1, \ldots, n_l + n_u.$$

Define intervals I_i's by

$$I_i = \{x \ : \ z_{i-1} < x < z_i\}, \ i = 0, 1, \ldots, n_l + n_u + 1$$

where we assume $z_{-1} = -\infty$ and $z_{n_l+n_u+1} = +\infty$.

The production rate in each I_i is a constant and denoted by u_i. We assume $u_0 = \bar{u}$ and $u_{n_l+n_u+1} = 0$ be fixed and in other regions $d < u_i \leq \bar{u}$. Let

$$A_i = \begin{pmatrix} \frac{\gamma+q_1(u_i)}{u_i-d} & -\frac{q(u_i)}{u_i-d} \\ \frac{q_2}{d} & -\frac{\gamma+q_2}{d} \end{pmatrix} \quad b_i = \begin{pmatrix} -\frac{1}{u_i-d} \\ \frac{1}{d} \end{pmatrix}$$

We have the following differential equations for the value functions:

$$\frac{dW(x)}{dx} = \begin{cases} A_i W(x) - b_i c^- x; & \text{if } x \in I_i,\ i = 0, \ldots, n_l \\ A_i W(x) + b_i c^+ x; & \text{if } x \in I_i,\ i = n_l + 1, \ldots, n_l + n_u + 1 \end{cases}$$

It is not difficult to solve this equation and the general solution is:

$$W(x) = \begin{cases} e^{A_0(x - z_0)}[W(z_0) - A_0^{-1} b_0 c^- z_0 - A_0^{-2} b_0 c^-] \\ \quad + A_0^{-1} b_0 c^- x + A_0^{-2} b_0 c^- & x \in I_0 \\ e^{A_i(x - z_{i-1})}[W(z_{i-1}) - A_i^{-1} b_i c^- z_{i-1} - A_i^{-2} b_i c^-] \\ \quad + A_i^{-1} b_i c^- x + A_i^{-2} b_i c^- & x \in I_i,\ i = 1, \ldots, n_l \\ e^{A_i(x - z_{i-1})}[W(z_{i-1}) + A_i^{-1} b_i c^+ z_{i-1} + A_i^{-2} b_i c^+] \\ \quad - A_i^{-1} b_i c^+ x - A_i^{-2} b_i c^+ & x \in I_i,\ i = n_l + 1, \ldots, n_l + n_u + 1 \end{cases}$$

Let $W(z_0) = A_0^{-1} b_0 c^+ z_0 + A_0^{-2} b_0 c^- + c\alpha_+$, where c is a constant to be decided, λ_+ is the unique positive eigenvalue of matrix A_0 and α_+ is the corresponding eigenvector. Given $W(z_0)$, $W(z_i)$ can be calculated recursively by continuity at z_i for $i = 1, 2, \cdots, n_l + n_u$. The last one calculated is $W(z_{n_l + n_u}) = E_1^{n_l + n_u} W(z_0) + \Sigma$ where $E_i^j = e^{A_i(z_i - z_{i-1})} \ldots e^{A_j(z_j - z_{j-1})}$ if $i \leq j$, and $E_i^j = I$ otherwise, and

$$\begin{aligned} \Sigma \;=\; & \sum_{j=1}^{n_l} E_{j+1}^{n_l + n_u} (A_j^{-1} b_j c^- z_j + A_j^{-2} b_j c^- - E_j^j A_j^{-1} b_j c^- z_{j-1} - E_j^j A_j^{-2} b_j c^-) \\ & + \sum_{j=n_l+1}^{n_l + n_u} E_{j+1}^{n_l + n_u} (E_j^j A_j^{-1} b_j c^+ z_{j-1} + E_j^j A_j^{-2} b_j c^+ - A_j^{-1} b_j c^+ z_j - A_j^{-2} b_j c^+) \end{aligned}$$

Using boundary condition, the constant c can be solved:

$$c = \frac{1}{\eta E_1^{n_l + n_u} \alpha_+} \left(c^+ z_{n_l + n_u} / q_1(d) - \eta E_1^{n_l + n_u} (A_0^{-1} b_0 c^- z_0 + A_0^{-2} b_0 c^-) - \eta \Sigma \right)$$

where $\eta = (1 + \gamma / q_1(d), -1)$. Thus we can evaluate value function $W(x)$ analytically for a given piecewise constant control policy described above.

Now let $y = (u_1, u_2, \ldots, u_{n_l + n_u}, z_{n_l + n_u})$. To emphasize the dependency of the value function to parameter y, we denote the value function by $W_i(x; y)$ $(i = 1, 2)$. Let S be the feasible region of y, then $S = \{y \,:\, d < u_i \leq \bar{u},\ i = 1, \ldots, n_l + n_u;\ z_{n_l + n_u} \geq 0\}$. Since each $y \in S$ correspond to a discretized control policy, we also call y control policy and S control policy set.

It is reasonable to think that when surplus x is large, we should let machine idle, and when x is very negative, we should produce at the maximum rate $\bar{u}$. Hence if we choose $-P_l$, P_u, n_l and n_u large enough, we can approximate any feasible control policy by a policy in S. In another words, the optimal control policy belongs to the closure of policy set S. Hence the solution to the nonlinear optimization $\min_{y \in S} W_i(x; y)$ gives us a control policy sufficiently closed to the optimal control policy. We shall give numerical examples in the following section.

5 Examples

In this section, we give three typical examples, which correspond to linear, convex and concave failure rate function respectively, to show how to find the optimal hedging points as well as the real optimal policies. To solve the numerical nonlinear optimization, we use optimization subroutine E04UCF of IBM NAG FORTRAN library. For all the examples in this section, we choose $n_l = n_u = 10$, $P_l = -3$, and failure function $q_1(u) = 0.5u^{\beta}$.

Example 1:

We already knew from [3] that if the failure rate function is linear then the hedging point policy is optimal and the optimal hedging point can be obtained by Equation (7) or (8). In this example we check the linear failure rate case by using numerical optimization procedure. We first calculate the optimal hedging point and the associated value function for various initial states. Then we run the optimization program and obtain the optimal control policy in S. Not surprisingly the results are quite consistent. For all initial states we chosen, the numerical results tell us that the hedging point policy is optimal and the optimal hedging point is $z = 1.01232$ which coincides with the analytical result.

The parameters are given as follows:

$$c^+ = 0.5, \quad c^- = 1.0, \quad \gamma = 0.1, \quad d = 1.0, \quad q_2 = 1.1, \quad \bar{u} = 2.0, \quad \beta = 1.0.$$

Example 2:

This example is for $\beta < 1$ case. We first explore the optimal hedging points for different states. For the cases which we only know the optimal hedging point is within certain interval, we use numerical method to find the optimal one. Then we explore the optimal policy using the nonlinear optimization procedure. The results show that the hedging point policy is still optimal but the optimal hedging points are dependent on the initial states. Both approaches give the same results. The optimal hedging points and the corresponding costs are given in Table 6.

The parameters we chosen are:

$$c^+ = 0.3, \quad c^- = 1.0, \quad \gamma = 0.1, \quad d = 1.0, \quad q_2 = 1.0, \quad \bar{u} = 2.0, \quad \beta = 0.5.$$

Example 3

In this example we check $\beta > 1$ case. As in the previous example, we first calculate optimal hedging points and the associated value functions and then compare the results with the real optimal control policy obtained by numerical optimization procedure. The numerical optimization results clearly show that the hedging point policy is not optimal in this case. All the optimal policies we find through optimization are the decreasing type controls. In other words, there exists an upper bound such that the production rate gradually reduces to the demand rate as the surplus approaches to that bound. Although the optimal controls may have slightly different upper bounds for different initial states, they are all similar in shapes. We only give two of them in Figure 1. The optimal hedging points and the corresponding costs as well as the optimal costs and upper bounds are shown in Table 7.

The parameters for this example are:

$$c^+ = 0.3, \quad c^- = 1.0, \quad \gamma = 0.1, \quad d = 1.0, \quad q_2 = 1.6, \quad \bar{u} = 2.0, \quad \beta = 1.5.$$

6 Conclusions

In this paper we developed an analytical and numerical combined method to derive the optimal hedging point policy as well as the optimal control policy for general production dependent failure functions. We analyzed the optimal hedging point and found that except for the linear failure rate function case the optimal hedging point is not unique and is dependent on the initial machine state and surplus state. We also solved analytically the value functions associate with a piecewise constant control policy and proposed a nonlinear programming to optimize the value functions numerically. We investigated three cases with failure rate function $q_1(u) = 0.5u^\beta$ for $\beta = 0.5, 1, 1.5$ respectively. As we expected, in the case of $\beta = 1$, the hedging point policy is optimal and all initial states have a unique optimal hedging point. In $\beta = 0.5$ case, the hedging point is still optimal but the optimal hedging point depend on the initial state. When $\beta = 1.5$, clearly the hedging point policy is not optimal and the optimal policy is a decreasing function of the surplus.

References

[1] R. Akella and P.R. Kumar. Optimal control of production rate in a failure prone manufacturing system. *IEEE Trans. on Automatic Control*, 31(2):116–126, Feb. 1986.

[2] T. Bielecki and P.R. Kumar. Optimality of zero-inventory policies for unreliable manufacturing systems. *Operations Research*, 36(4):532–541, July-Aug. 1988.

[3] J.-Q. Hu, P. Vakili, and G.-X. Yu. Optimality of hedging point policies in the production control of failure prone manufacturing systems. to appear in *IEEE Trans. on Automatic Control*, 1991.

[4] J.Q. Hu and D. Xiang. The queueing equivalence to a manufacturing system with failures. *IEEE Trans. on Automatic Control*, 38(3):499–502, March 1993.

[5] J.G. Kimemia and S.B. Gershwin. An algorithm for the computer control of production in flexible manufacturing systems. *IIE Transaction*, 15:353–362, Dec. 1983.

[6] G. Liberopoulos and M. Caramanis. Production control of manufacturing systems with production rate dependent failure rates. to appear in *IEEE Trans. on Automatic Control*, 1992.

[7] A. Sharifnia. Optimal production control of a manufacturing system with machine failures. *IEEE Trans. on Automatic Control*, 33(7):620–625, July 1988.

[8] G.-X. Yu. *Flow control of stochastic manufacturing systems.* PhD thesis, Boston University, 1993. in preparation.

$x < z_1$	$z_1 < x < z_2$		$x > z_2$	
$\alpha = 1$ or $\alpha = 2$	$\alpha = 1$	$\alpha = 2$	$\alpha = 1$	$\alpha = 2$
z_1	x	(z_1, x)	z_2	(z_1, z_2)

Table 1: Location of the optimal hedging point for case $z_2 > z_1 > 0$

$x < z_1$	$z_1 < x < z_2$		$x > z_2$	
$\alpha = 1$ or $\alpha = 2$	$\alpha = 1$	$\alpha = 2$	$\alpha = 1$	$\alpha = 2$
0	x^+	$[0, x^+)$	z_2	$(0, z_2)$

Table 2: Location of the optimal hedging point for case $z_2 > 0 > z_1$.

$x < z_2$	$z_2 < x < z_1$		$x > z_1$	
$\alpha = 1$ or $\alpha = 2$	$\alpha = 1$	$\alpha = 2$	$\alpha = 1$	$\alpha = 2$
z_1	$\arg\min_{z \in \{z_1, z_2\}} W_1^z(x)$	$\arg\min_{z \in (z_2, x) \cup \{z_1\}} W_2^z(x)$	z_2	(z_2, z_1)

Table 3: Location of the optimal hedging point for case $z_1 > z_2 > 0$

$x < z_2$	$z_2 < x < z_1$		$x > z_1$	
$\alpha = 1$ or $\alpha = 2$	$\alpha = 1$	$\alpha = 2$	$\alpha = 1$	$\alpha = 2$
z_1	$\arg\min_{z \in \{0, z_2\}} W_1^z(x)$	$\arg\min_{z \in (0, x) \cup \{z_1\}} W_2^z(x)$	0	$(0, z_1)$

Table 4: Location of the optimal hedging point for case $z_1 > 0 > z_2$

machine state	surplus	hedging point	cost
1	-2.0	1.01232	19.371195
2	-2.0	1.01232	25.739853
1	-3.0	1.01232	26.459651
2	-3.0	1.01232	33.764950
1	2.0	1.01232	9.415612
2	2.0	1.01232	9.693503
1	3.0	1.01232	9.713095
2	3.0	1.01232	9.796794

Table 5: Numerical results for $\beta = 1.0$

machine state	surplus	hedging point	cost
1	-2.0	1.52305	14.1974
2	-2.0	1.52305	20.6851
1	-3.0	1.52305	20.0289
2	-3.0	1.52305	27.7143
1	2.0	1.44034	6.7140
2	2.0	1.48466	6.9810
1	3.0	1.44034	6.7912
2	3.0	1.45474	6.8803
1	1.48	1.52305	6.7930
2	1.48	1.52302	7.2644
1	1.5	1.44034	6.7887
2	1.5	1.52305	7.2495

Table 6: Numerical results for $\beta = 0.5$

initial state		hedging point policy		numerical optimization	
machine state	surplus	hedging point	cost	upper bound	value function
1	-2.0	1.1363	17.277340	1.2368	17.259644
2	-2.0	1.1363	21.997956	1.2368	21.984925
1	-3.0	1.1363	24.845609	1.2372	24.834581
2	-3.0	1.1363	30.207316	1.2369	30.199195
1	2.0	1.2563	6.000400	1.3699	5.954256
2	2.0	1.2308	6.130749	1.3660	6.084606
1	3.0	1.2563	6.145484	1.3693	6.103732
2	3.0	1.2516	6.169372	1.3693	6.127545
1	1.2	1.2	6.101946	1.2389	6.062292
2	1.2	1.1499	6.600400	1.2377	6.555229

Table 7: Numerical results for $\beta = 1.5$

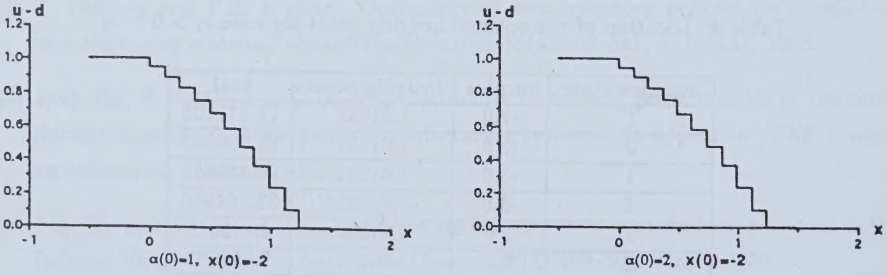


Figure 1: Optimal control policies for $\beta = 1.5$

Finite Horizon Production Planning of Stochastic Manufacturing Systems: Optimality, Traceability and Turnpike Sets[1]

Q. Zhang and G. Yin

Department of Mathematics, University of Kentucky
Lexington, KY 40506

Department of Mathematics, Wayne State University
Detroit, MI 48202

Abstract. This work is concerned with finite horizon production planning of stochastic manufacturing systems. The objective is to derive a production plan that minimizes an expected discounted cost of inventories/backlogs. Our aim is to derive explicit solutions to the problems under consideration. Owing to the finite horizon formulation, the associated turnpike sets are time dependent curves. It is shown that the optimal controls can be written in terms of these turnpike sets.

Keywords. stochastic production planning, finite horizon, turnpike sets, traceability.

1 Introduction

This work is concerned with finite horizon production planning of stochastic manufacturing systems. Two problems will be considered. In the first problem, a manufacturing system with a machine subject to breakdown and constant product demand rate is considered, and in the second problem, a manufacturing system with constant production capacity and increasing (random jump) product demand rate is dealt with. In these problems, by taking the difference of the real and planned cumulative goods of the finished ones as the states of the system, the objective is to choose admissible production rates to minimize the inventory/backlog costs over a finite horizon. Since a large amount of production planning concerns with finite horizon, and since the infinite horizon problem can be considered as limits of the corresponding finite time problems, the study of finite horizon problems is not only worthwhile, but also necessary.

Recently, production planning for manufacturing systems has received much attention. Related literature can be found in Kimemia and Gershwin [8], Akella and Kumar [1], Bielecki and Kumar [2], Sethi *et al.* [11], and Hu and Xiang [7]. In these infinite horizon problems, one of the important tasks appears to be the utilization of turnpike sets.

It is the purpose of this work to study the turnpike sets with finite time horizon. It should be pointed out that there are some distinct differences between the finite time and the infinite horizon formulations. For an infinite horizon formulation, such as the problems studied in [1] and [2], the dynamics of the systems are essentially homogeneous,

---

[1] This research was supported in part by the National Science Foundation under Grant DMS-9224372.

and as a consequence, the turnpike sets consist of constants, which completely characterize the optimal control policies. If the system performance is evaluated over a finite time horizon, the threshold levels are no longer constants, but "time dependent threshold curves". Therefore, the problem becomes much more complicated.

By considering relatively simple models, we aim at deriving explicit optimal solutions. It will be shown that there exist turnpike sets and the optimal control can be determined explicitly by such inventory levels. For both the problem with random machine capacity and constant demand, and the problem with constant machine capacity and random demand, the explicit solutions of the optimal controls are obtained. Proofs of results are omitted and referred to [15].

2 Problem 1: Constant product demand

Let $x_t \in R^1$ denote the inventory/backlog process (state variable) and $u_t \geq 0$ denote the rate of production planning (control variable) of a manufacturing system. The product demand is assumed to be a constant and denoted by z. Then,

$$\dot{x}_t = u_t - z, \; x_s = x, \; 0 \leq s \leq t \leq T \tag{2.1}$$

where T is a finite number.

Let $\mathcal{M} = \{u_1, u_2\}$ ($u_1 > u_2 \geq 0$) denote the set of machine states and let $\alpha(t) \in \mathcal{M}$ denote the machine capacity process in the manufacturing system. If $\alpha(t) = u_1$, the machine is in a good condition with capacity u_1. If $\alpha(t) = u_2$, the machine (or part of the machine) breaks down with a remaining capacity u_2. We always assume that $u_1 > z > u_2$, i.e., the demand can be satisfied if the machine is in a good condition and cannot be satisfied if the machine (or part of the machine) breaks down.

Let the cost function $J(s, x, \alpha, u.)$ with $\alpha(s) = \alpha \in \mathcal{M}$ be defined by

$$J(s, x, \alpha, u.) = E \int_s^T e^{-\rho t} h(x_t) dt, \tag{2.2}$$

where $\rho \geq 0$ is the discount factor and $h(x)$ is the running cost. The problem is to find a production plan $0 \leq u_t \leq \alpha(t)$ that minimizes $J(s, x, \alpha, u.)$.

Assumptions.

(A1) $h(x)$ is a convex function such that for positive constants C_h and k_h,

$$0 \leq h(x) \leq C_h(1 + |x|^{k_h}) \text{ and } h(x) > h(0) = 0 \text{ for all } x \neq 0.$$

Moreover, there exists a constant $c_h > 0$ such that

$$\frac{h_{x^+}(x_2) - h_{x^+}(x_1)}{x_2 - x_1} \geq c_h \text{ for all } -|u_2 - z|T \leq x_1 < 0 \leq x_2 \leq |u_2 - z|T, \tag{2.3}$$

where $h_{x^+}(x)$ denotes the right-hand derivative of $h(x)$.

(A2) The capacity process $\alpha(t) \in \mathcal{M}$ is a two-state Markov chain governed by

$$L_\alpha f(\cdot)(i) = \begin{cases} \lambda(f(u_2) - f(u_1)) & \text{if } i = u_1 \\ 0 & \text{if } i = u_2. \end{cases} \tag{2.4}$$

for any function f on $\mathcal{M}$, where $\lambda \geq 0$ is the machine breakdown rate, and the repair rate μ is assumed to be 0.

Examples of $h(x)$. Let us give a few examples of $h(x)$ that satisfy Assumption (A1).

(1) $h(x) = x^2$;

(2) $h(x) = h^+ \max\{0, x\} + h^- \max\{0, -x\}$ where $h^+ > 0$ and $h^- > 0$ are constants. This cost function was employed in [1].

(3) $h(x)$ is convex and piecewise linear with $h(0) = 0$, $h_{x^-}(0) < 0$, and $h_{x^+}(0) > 0$.

Let $v(s, x, \alpha)$ denote the value function of the problem. We modify the turnpike definition given in [11] to incorporate the variation of the turnpike sets with the changes of time.

Definition. $\phi(s)$ and $\psi(s)$ are said to be the *turnpike sets* for $\alpha = u_1$ and $\alpha = u_2$, respectively, if for all $s \in [0, T]$,

$$\begin{aligned} & v(s, \phi(s), u_1) = \min_x v(s, x, u_1) \\ \text{and} \quad & v(s, \psi(s), u_2) = \min_x v(s, x, u_2), \text{ respectively.} \end{aligned}$$

In view of the Verification Theorem (cf. [4]), the optimal feedback control should be a function of (s, x, α) that minimize $(u - z)v_x(s, x, \alpha)$. Note that if v is differentiable, then $v_x(s, \phi(s), \alpha_1) = 0$ and $v_x(s, \psi(s), \alpha_2) = 0$. Since $v(s, \cdot, \alpha)$ is convex,

$$\begin{aligned} v_x(s, x, \alpha_1) &\geq 0 \text{ if } x \geq \phi(s) \\ v_x(s, x, \alpha_1) &\leq 0 \text{ if } x \leq \phi(s) \\ v_x(s, x, \alpha_2) &\geq 0 \text{ if } x \geq \psi(s) \\ v_x(s, x, \alpha_2) &\leq 0 \text{ if } x \leq \psi(s). \end{aligned}$$

In the sequel, we shall show that $\phi(s)$ and $\psi(s)$ are absolute continuous and satisfy $0 \leq \dot{\phi}(s) + z \leq \alpha_1$ and $\dot{\psi}(s) + z > \alpha_2$. Therefore, the optimal feedback control $u^*(t, x, \alpha)$ should have the following form:

$$\text{If } \alpha(t) = \alpha_1, \text{ then } \ u^*(t, x, \alpha_1) = \begin{cases} 0 & \text{if } x > \phi(t) \\ \dot{\phi}(t) + z & \text{if } x = \phi(t) \\ \alpha_1 & \text{if } x < \phi(t). \end{cases} \tag{2.5}$$

$$\text{If } \alpha(t) = \alpha_2, \text{ then } \ u^*(t, x, \alpha_2) = \begin{cases} 0 & \text{if } x > \psi(t) \\ \alpha_2 & \text{if } x \leq \psi(t). \end{cases} \tag{2.6}$$

Moreover, it is easy to see that under the control policy $u^*_t = u^*(t, x_t, \alpha(t))$, the ordinary differential equation

$$\dot{x}^*_t = u^*(t, x^*_t, \alpha(t)) - z,\ x^*_s = x$$

has a unique solution.

By virtue of the transformation $\tilde{v}(s, x, \alpha_1) = e^{-\lambda s} v(s, x, \alpha_1)$, the problem we wish to solve is equivalent to:

$$\begin{aligned} &\text{min.} && \int_s^T e^{-\lambda(t-s)}[e^{-\rho t} h(x_t) + \lambda v(t, x_t, \alpha_2)]dt \\ &\text{s.t.} && \dot{x}_t = u_t - z, x_s = x,\ 0 \le u_t \le \alpha_1, \end{aligned} \tag{2.7}$$

where $v(t, x_t, \alpha_2)$ can be obtained from the following problem:

$$\begin{aligned} &\text{min.} && \int_s^T e^{-\rho t} h(x_t) dt \\ &\text{s.t.} && \dot{x}_t = u_t - z, x_s = x,\ 0 \le u_t \le \alpha_2. \end{aligned} \tag{2.8}$$

In order to carry out the analysis, we need to evaluate $v(s, x, \alpha_2)$. If $\alpha(s) = \alpha_2$, by virtue of Assumption (A2) on $\alpha(t)$, $\alpha(t) \equiv \alpha_2$ for $t \in [s, T]$. Therefore, the problem becomes a deterministic problem. If α_2 were larger than z, the problem would become much simpler: one would only need to produce at the maximum speed if $x_t < 0$ and produce nothing if $x_t > 0$. However, in our case, such production policy is no longer optimal since the state x_t cannot stay at $x = 0$. Intuitively, if the initial value $x_s \le 0$, then it is better to take $u_t = \alpha_2$ in order to keep x_t (≤ 0) as close as possible to $x = 0$ from below. If $x_s \ge z(T - s)$, then it is better to take $u_t = 0$ so as to keep x_t (≥ 0) as close as possible to $x = 0$ from above. Let us now consider the case when x_s is less than $z(T - s)$ but is close to $z(T - s)$. In this case, if we take $u_t = 0$ at the beginning for a while, then switch the control to $u_t = \alpha_2$ before x_t actually reaches 0. It can be seen that in this way the cost will be smaller than the cost incurred if we use $u_t = 0$ until $x_t = 0$, then switch to $u_t = \alpha_2$. Therefore, there must be a curve, say $\psi(s) \ge 0$ (which will be seen to be a turnpike set), such that $u_t = 0$ when $x_t \ge \psi(t)$ and $u_t = \alpha_2$ when $x_t < \psi(s)$. Based on these observations, a crucial task is to identify this curve $\psi(s)$. For infinite horizon formulation, usually turnpike sets are attractors for optimal trajectories, whereas in the current setting, the turnpike sets are switching curves.

With the observation and heuristics above, we are now ready to state the following:

Theorem 1. *Let $\phi(s)$ and $\psi(s)$ be the turnpike sets for $\alpha = u_1$ and $\alpha = u_2$, respectively. Then $\phi(s)$ and $\psi(s)$ are single-valued absolutely continuous functions. They are uniquely determined by*

$$h(\phi(s)) + \lambda e^{\rho s} v(s, \phi(s), u_2) = \min_x [h(x) + \lambda e^{\rho s} v(s, x, u_2)] \tag{2.9}$$

$$\text{and } \int_s^T e^{-\rho t} h_{x^+}(\psi(s) + (u_2 - z)(t - s))dt = 0. \tag{2.10}$$

Furthermore,

(1) $0 < \psi(s) < |u_2 - z|(T-s)$ *for* $s \in [0,T)$ *and* $\psi(T) = 0$;

(2) $z \geq \dot{\psi}(s) + z > u_2$, *a.e. in* $s \in [0,T]$.

(3) $0 \leq \phi(s) < \psi(s)$ *for* $s \in [0,T)$ *and* $\phi(T) = 0$;

(4) $0 \leq \dot{\phi}(s) + z \leq z$, *a.e. in* $s \in [0,T]$.

(Any absolutely continuous function $\gamma(s)$ *that satisfies (4) is said to be traceable.)*

Let $u^*(t,x,\alpha)$ *be defined as follows:*

$$u^*(t,x,\alpha) = \begin{cases} u^*(t,x,u_1) = \begin{cases} 0 & \text{if } x > \phi(t) \\ \phi(t) + z & \text{if } x = \phi(t) \\ u_1 & \text{if } x < \phi(t). \end{cases} & \text{if } \alpha = u_1 \\ u^*(t,x,u_2) = \begin{cases} 0 & \text{if } x > \psi(t) \\ u_2 & \text{if } x \leq \psi(t). \end{cases} & \text{if } \alpha = u_2. \end{cases} \tag{2.11}$$

Under the control policy $u_t^* = u^*(t, x_t, \alpha(t))$, *the system*

$$\dot{x}_t^* = u^*(t, x_t^*, \alpha(t)) - z, \ x_s^* = x$$

has a unique solution. The feedback control $u_t^* = u^*(t, x_t, \alpha(t))$ *is optimal.*

Example 1. If the cost function $h(x) = h^+ \max\{0, x\} + h^- \max\{0, -x\}$, then, by using (2.9) and (2.10), we find

$$\phi(s) = \max\{0, -\frac{|u_2 - z|}{\rho} \log \frac{\rho h^+ + \lambda(h^+ + h^- e^{-\rho(T-s)})}{\lambda(h^+ + h^-)}\} \tag{2.12}$$

$$\text{and } \psi(s) = \frac{u_2 - z}{\rho} \log \frac{h^+ + h^- e^{-\rho(T-s)}}{h^+ + h^-}.$$

Equivalently, $\phi(s)$ can also be written as:

$$\phi(s) = 0 \text{ for all } s \in [0,T] \text{ if } \lambda h^- - \rho h^+ \leq 0;$$

otherwise,

$$\phi(s) = \begin{cases} -\frac{|u_2 - z|}{\rho} \log \frac{\rho h^+ + \lambda(h^+ + h^- e^{-\rho(T-s)})}{\lambda(h^+ + h^-)} & \text{if } s \leq T + \rho^{-1} \log \frac{\lambda h^- - \rho h^+}{\lambda h^-} \\ 0 & \text{if } s > T + \rho^{-1} \log \frac{\lambda h^- - \rho h^+}{\lambda h^-} \end{cases}.$$

As $T \to \infty$, it is easily seen that

$$\phi(s) \to \max\{0, -\frac{|u_2 - z|}{\rho} \log \frac{(\rho h^+ + \lambda h^+)}{\lambda(h^+ + h^-)}\},$$

which gives the same turnpike set as in [1] provided that the repair rate is 0.

3 Problem 2: Constant machine capacity

The second problem is a slight variation of the first one. In this problem, the system having a constant machine capacity $u_0 > 0$ with a random demand rate z_t, is described by the following equation

$$\dot{x}_t = u_t - z_t,\ x_s = x,\ 0 \le s \le t \le T < \infty \tag{3.1}$$

with the production constraints $0 \le u_t \le u_0$.

Let $\mathcal{Z} = \{z_1, z_2\}$ denote the set of demand rates with $0 < z_1 < u_0 < z_2$.

The corresponding cost function $J(s, x, z, u.)$ with $z_s = z \in \mathcal{Z}$ is defined by

$$J(s, x, z, u.) = E \int_s^T e^{-\rho t} h(x_t) dt. \tag{3.2}$$

Similar to the previous section, the problem is to find a production plan $0 \le u_t \le u_0$ that minimizes $J(s, x, z, u.)$.

Assumptions.

(A1') Let Assumption (A1) be satisfied with (2.3) replaced by

$$\frac{h_{x^+}(x_2) - h_{x^+}(x_1)}{x_2 - x_1} \ge c_h \text{ for all } -|u_0 - z_2|T \le x_1 < 0 \le x_2 \le |u_0 - z_2|T,$$

with z_2 given below.

(A2') The demand process $z_t \in \mathcal{Z}$ is also a two state Markov chain governed by

$$L_z f(\cdot)(i) = \begin{cases} \lambda'(f(z_2) - f(z_1)) & \text{if } i = z_1 \\ 0 & \text{if } i = z_2, \end{cases} \tag{3.3}$$

for any function f on $\mathcal{Z}$.

Let $v(s, x, z)$ denote the value function of the problem.

Definition. $\phi(s)$ and $\psi(s)$ are said to be *turnpike sets* if

$$v(s, \phi(s), z_1) = \min_x v(s, x, z_1)$$
$$\text{and} \quad v(s, \psi(s), z_2) = \min_x v(s, x, z_2).$$

Theorem 2. *Suppose (A1') and (A2') and $z_1 \ge z_2 - u_0$. Then, the turnpike sets $\phi(s)$ and $\psi(s)$ are single-valued absolute continuous functions. They are uniquely determined by:*

$$\begin{aligned} & h(\phi(s)) + \lambda \textstyle\int_s^T e^{-\rho t} h(\phi(s) + (u_0 - z_2)(t - s)) dt \\ = \ & \min_x h(x) + \lambda \textstyle\int_s^T e^{-\rho t} h(x + (u_0 - z_2)(t - s)) dt \end{aligned} \tag{3.4}$$

$$\int_s^T e^{-\rho t} h_{x^+}(\psi(s) + (u_0 - z_2)(t - s)) dt = 0, \tag{3.5}$$

and satisfy the following properties:

(1) $0 < \psi(s) < |u_0 - z_2|(T-s)$ *for* $s \in [0,T)$ *and* $\psi(T) = 0$;

(2) $z_2 \geq \dot{\psi}(s) + z_2 > u_0$;

(3) $0 \leq \phi(s) < \psi(s)$ *for* $s \in [0,T)$ *and* $\phi(T) = 0$;

(4) $z_1 \geq \dot{\phi}(s) + z_1 \geq u_0 - z_2 + z_1 \geq 0$.

Let $u^*(t, x_t, z)$ *be defined as follows:*

$$u^*(t,x,z) = \begin{cases} u^*(t,x,z_1) = \begin{cases} 0 & \text{if } x > \phi(t) \\ \dot{\phi}(t) + z_1 & \text{if } x = \phi(t) \\ u_0 & \text{if } x < \phi(t) \end{cases} \\ u^*(t,x,z_2) = \begin{cases} 0 & \text{if } x > \psi(t) \\ u_0 & \text{if } x \leq \psi(t). \end{cases} \end{cases} \tag{3.6}$$

Then under the control $u_t^* = u^*(t, x_t, z_t)$, *the equation*

$$\dot{x}_t^* = u^*(t, x_t^*, z_t) - z_t, \; x_0^* = x$$

has a unique solution. Therefore, the control u_t^* *is optimal.*

Example 2. We consider the cost $h(x) = h^+ \max\{0, x\} + h^- \max\{0, -x\}$. Then, we can use (3.4) and (3.5) to find

$$\phi(s) = \max\{0, -\frac{|u_2 - z|}{\rho} \log \frac{\rho h^+ + \lambda(h^+ + h^- e^{-\rho(T-s)})}{\lambda(h^+ + h^-)}\}.$$

$$\psi(s) = \frac{u_0 - z_2}{\rho} \log \frac{h^+ + h^- e^{-\rho(T-s)}}{h^+ + h^-}.$$

Note that the assumption $z_1 \geq z_2 - \alpha_0$ in Theorem 4.1 is a relatively conservative one. In Example 2, this condition can be relaxed to

$$z_1 \geq |\alpha_0 - z_2| \frac{\lambda h^- - \rho h^+}{\lambda(h^+ + h^-)}. \tag{3.7}$$

It can be seen that (3.7) is also necessary for the traceability of $\phi(s)$ in this example. If (3.7) fails. $\phi(s)$ will no longer be the turnpike of the problem since it is not traceable on $[0,T]$. Let $\tilde{\phi}(s)$ denote the turnpike set for $\alpha = \alpha_1$ defined as $v(s, \tilde{\phi}(s), z_1) = \min_x v(s, x, z_1)$. In fact, $0 \leq \tilde{\phi}(s) \leq \psi(s)$. It can be shown that $h(x) + \lambda v(s, x, z_2)$ is strictly convex on $[0, \psi(s)]$, which implies that $v(s, x, z_1)$ is also strictly convex on $[0, \psi(s)]$ (see [11] for details). Therefore, $\tilde{\phi}(s)$ is a continuous function. Intuitively, the optimal control $u^*(t, x, \alpha_1)$ should be given as in (3.6) with $\tilde{\phi}(s)$ in place of $\phi(s)$ provided that $\tilde{\phi}(s)$ is traceable. Let $s_0 = T + \rho^{-1} \log \dfrac{\lambda h^- - \rho h^+}{\lambda h^-}$. Then, $0 \leq \dot{\phi}(s) + z_1 \leq \alpha_0$ for $s \geq s_0$. This implies $\tilde{\phi}(s) = \phi(s)$ for $s \geq s_0$. However, if $0 > s_0 > T$ and z_2 is large enough, the traceability property of $\tilde{\phi}(s)$ will not hold, which makes the problem very complicated; explicit optimal solution is very difficult to obtain.

4 Further remarks

We would like to say a few words about Problem 1 in which repair facilities are available ($\mu > 0$). In this case, the machine capacity process $\alpha(t)$ is allowed to jump back and forth. The explicit optimal control of the resulting problem is very difficult to obtain. However, instead of searching for exact optimal solutions, one may be interested in finding approximate optimal solutions. If $\alpha(t)$ is assumed to be a finite state Markov chain (cf. [1] and [11]), then there exists a sequence of exponential random variables where the jumps of $\alpha(t)$ takes place. One may, therefore, employ the idea used in [9] to write the value function v as the limit of a sequence of value functions $v^{(n)}$ in which only n jumps is allowed in the finite horizon under consideration. In this way, the original stochastic control problem can be approximated by a sequence of deterministic control problems. For each n one may also define curves $\phi(s)$ and $\psi(s)$ similarly as in Theorem 1. The traceability of these curves are again difficult to verify analytically. One has to resort to numerical methods for solutions. Some previous discussions on numerical solutions for production planning problems can be found in [3] and [8] among others. A different approach using stochastic optimization type of recursive algorithms (via perturbation analysis) is contained in [14] and [13].

References

[1] R. Akella and P.R. Kumar, Optimal control of production rate in a failure-prone manufacturing system, *IEEE Trans. Automat. Control* **AC-31** (1986), pp. 116-126.

[2] T. Bielecki and P.R. Kumar, Optimality of zero-inventory policies for unreliable manufacturing systems, *Oper. Res.* **26** (1988), pp. 532-546.

[3] K. Boukas and A. Haurie, Planning production and preventive maintenance in a flexible control approach, *Proc. 27th IEEE Conf. Decision & Control*, 1988, pp. 2294-2300.

[4] W.H. Fleming and R.W. Rishel, *Deterministic and Stochastic Optimal Control.* Springer-Verlag, New York, 1975.

[5] W.H. Fleming, S.P. Sethi and H.M. Soner, An optimal stochastic production planning problem with randomly fluctuating demand, *SIAM J. Control Optim.* **25** (1987), pp. 1494-1502.

[6] W.H. Fleming and H.M. Soner, *Controlled Markov Processes and Viscosity Solutions*, Springer-Verlag, New York, 1992.

[7] J.Q. Hu and D. Xiang, Structural properties of optimal production controllers in failure prone manufacturing systems, preprint, (1992).

[8] J.G. Kimemia and S.B. Gershwin, An algorithm for the computer control production in flexible manufacturing systems, *IIE Trans.* **15** (1983), pp. 353-362.

[9] R.W. Rishel, Dynamic programming and minimum principles for systems with jump Markov disturbances, *SIAM J. Control Optim.* **13** (1974), pp. 338-371.

[10] R.W. Rishel, Control of systems with jump Markov disturbances, *IEEE Trans. Automat. Control* **AC-20** (1975), pp. 241-244.

[11] S.P. Sethi, H.M. Soner, Q. Zhang, and J. Jiang, Turnpike sets and their analysis in stochastic production planning problems, forthcoming in *Math. Oper. Res.* (1992).

[12] H.M. Soner, Optimal stochastic control with state-space constrains, II, *SIAM J. Control Optim.* **24** (1986), pp. 1110-1123.

[13] H.M. Yan, G. Yin and X. Lou, Using stochastic optimization to determine threshold values for control of unreliable manufacturing systems, to appear in *J. Optim. Theory Appl.* (1993).

[14] G. Yin, H.M. Yan and X. Lou, On a class of stochastic optimization algorithms with applications to manufacturing models, *Proc. The Third Model Oriented Data Analysis*, Physica-Verlag.

[15] Q. Zhang and G. Yin, Turnpike sets in stochastic manufacturing systems with finite horizon, Institute for Mathematics and its Applications, University of Minnesota, IMA Preprint Series #1129, (1993).

HUMAN VISUAL SYSTEM BASED SUBBAND IMAGE COMPRESSION

Thomas P. O'Rourke and Robert L. Stevenson
Laboratory for Image and Signal Analysis
Department of Electrical Engineering
University of Notre Dame
Notre Dame, Indiana 46556

ABSTRACT

In this paper we propose a wavelet-based image compression technique which incorporates some properties of the human visual system (HVS). A new wavelet decomposition based on the HVS will be introduced. The wavelet coefficients will be quantized with a vector quantization (VQ) technique which uses HVS properties to allocate bits to the various subbands. It will be shown that compression techniques based on HVS considerations yield higher quality results compared to standard techniques. Experimental results will be shown.

1 INTRODUCTION

Due to limitations in bandwidth or storage space, many applications require compression of digital images. The current state-of-the-art image compression scheme is the Joint Photographic Experts Group (JPEG) Standard [1, 2]. One problem with this scheme is the appearance of blocking artifacts in the reconstructed image which are often highly noticeable to a human observer. The use of the wavelet transform for image compression was suggested in [3] using a separable extension of one-dimensional (1-D) wavelet theory to two dimensions. The use of compactly supported wavelets, introduced in [4], greatly reduces the computational cost of the wavelet transform. A non-separable extension to 2-D is discussed in [5], although a number of challenges in the design remain unresolved. Image compression involving the wavelet transform is discussed in [6, 7] and in combination with VQ in [8]. In this paper, we examine a new HVS based wavelet subband decomposition in combination with VQ using HVS based bit allocation. The results are compared with other wavelet subband approaches as well as with JPEG.

2 RELEVANT HUMAN VISUAL SYSTEM PROPERTIES

This brief overview of relevant information on the human visual system (HVS) is based on research done in the 1960's by Campbell and others [9]. In the proposed compression technique, two important characteristics of the HVS will be incorporated into the compressed data representation. These properties are the orientation sensitivity and the contrast sensitivity of the HVS. The findings of [9] were based on psychophysical experiments in which subjects determined the threshold where a sinusoidal grating was just noticeable.

In an experiment concerning the sensitivity of the HVS to frequencies at various spatial orientations [9], it was found that the HVS could better resolve frequencies along the horizontal and vertical orientation than along the diagonal orientations. This can be seen in Figure 1 which plots the point for which the spatial frequency at each orientation can no longer be resolved by the HVS. Notice that this experimental data [9] is roughly

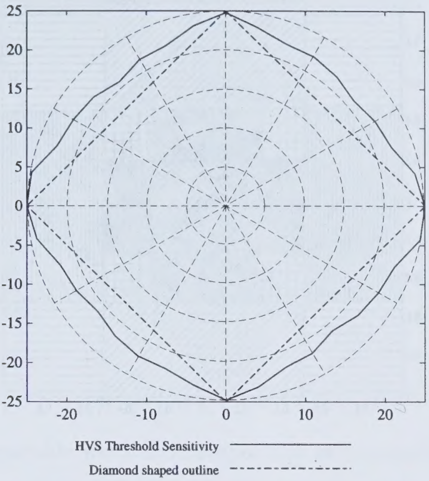


Figure 1: Resolvability threshold for sinusoidal gratings, cycles per degree.

diamond shaped as evident by the diamond which is also plotted on the graph. This implies that the passband of the subband coding system should be diamond shaped to take advantage of this HVS characteristic.

In the same set of experiments [9], the contrast sensitivity of the HVS was also examined. For spatial frequencies above 7 cycles per degree, it was found that the logarithm of the contrast sensitivity is a linear function of spatial frequency. The slope of this linear function was found to be different for horizontal and vertical orientations relative to the oblique 45° and 135° orientations. The log sensitivity approximately satisfies

$$\log S(f) = \log 500(1 + \frac{1}{-50}f) \tag{1}$$

for the vertical and horizontal orientation and

$$\log S(f) = \log 500(1 + \frac{1}{-40}f) \tag{2}$$

for the oblique orientation, where f is the spatial frequency in cycles per degree [9].

The HVS has a diamond shaped frequency passband which will be exploited in the design of the proposed image compression scheme. Additionally, the logarithm of the contrast sensitivity is approximated by a linear function which will be used for bit allocation. The perceived quality of the reconstructed images will be improved by consideration of the HVS in the design of the image compression system.

3 TWO-DIMENSIONAL WAVELET SUBBAND DECOMPOSITIONS

The wavelet transform can be considered a special case of perfect reconstruction subband coding with additional regularity requirements on the filters[10]. The idea behind subband coding is to decompose a signal $x(n)$ into two or more bandpass signals, $w_i(n)$,

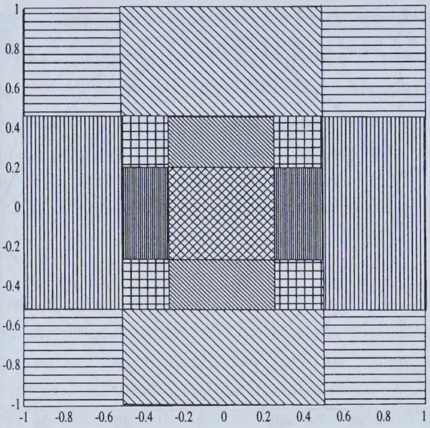


Figure 2: Separable wavelet decomposition of frequency space, 2 levels.

which are coded and transmitted individually over a channel. Since each bandpass signal has its frequency content in a certain range, knowledge about the frequency response of a human observer can be used in the coding of each bandpass signal [11]. More error can be tolerated at frequencies where the observer has lower sensitivity to noise. See [12, 13] for more information on image subband coding.

3.1 Separable Wavelets

The FIR filters used are separable products of 1-D filters taken from a 1-D wavelet decomposition. Figure 2 shows how the separable wavelet transform decomposes the signal frequency space. Due to the separability of the filters, the wavelet transform can be calculated at each level by filtering first in one dimension and then in the other. The original image can be recovered exactly from its wavelet decomposition provided the transform coefficients are not quantized. The separable wavelet transform is the most common method for extending the wavelet transform to 2-D.

3.2 Non-separable Wavelets

Figure 3 illustrates the decomposition of the signal frequency space when the low-pass branch of the non-separable subband system is iterated. The sampling pattern of the subband signals alternates between rectangular and quincunx. From [5], there exists a non-separable low-pass FIR filter which leads to a compactly supported scaling function and orthonormal compactly supported wavelets. The regularity of the scaling function cannot be increased by simply increasing the filter order as was the case with the 1-D wavelet system. This limitation is avoided in other approaches in [5], but each of those also has major disadvantages.

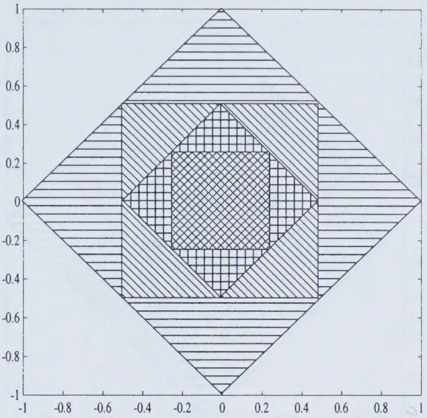


Figure 3: Non-separable wavelet decomposition of frequency space, 4 levels.

3.3 HVS-based Wavelets

The first stage of this decomposition uses a filter with a diamond shaped 2-D frequency response. The input is rectangularly sampled and the output is quincunx sampled. This stage is the same as the first stage in the decomposition in section 3.2. The regularity of this filter does not matter because this stage is not repeated. The next stage and all following stages use a 4-band decomposition similar to the one used in section 3.1.

The quincunx sampling pattern can be viewed as a rectangular sampling pattern which has been rotated 45° and scaled by $\sqrt{2}$. The new coordinates for the rotated sampling pattern in terms of the rectangular sampling pattern coordinates can be found by

$$\begin{bmatrix} n_1 \\ n_2 \end{bmatrix}' = \begin{bmatrix} \sqrt{2} & 0 \\ 0 & \sqrt{2} \end{bmatrix} \begin{bmatrix} \cos 45^\circ & -\sin 45^\circ \\ \sin 45^\circ & \cos 45^\circ \end{bmatrix} \begin{bmatrix} n_1 \\ n_2 \end{bmatrix} = \begin{bmatrix} n_1 - n_2 \\ n_1 + n_2 \end{bmatrix} \quad (3)$$

In the frequency domain, this amounts to a rotation by -45° and a scaling by $\frac{1}{\sqrt{2}}$. The new HVS based frequency decomposition is shown in Figure 4. The diamond shaped subbands are meant to fit the diamond shaped sensitivity of the HVS shown in Figure 1.

This HVS decomposition has the advantage over the non-separable diamond shaped wavelet because the regularity of the wavelet can be increased by increasing the order of the 1-D filters from which it is derived. The regularity of the non-separable wavelets cannot easily be increased. The HVS based decomposition is separable along the 45° and 135° diagonals and has all the advantages of the separable decomposition. The HVS based decomposition has an advantage over standard separable decompositions because it is oriented to fit the diamond shaped HVS sensitivity.

4 WAVELETS FOR IMAGE COMPRESSION

The quantization technique used here is vector quantization (VQ) [14]. In this section, the bit allocation method and VQ of transform coefficients are discussed. The three wavelet decompositions from Section 3 are quantized and the resulting reconstructed images are compared.

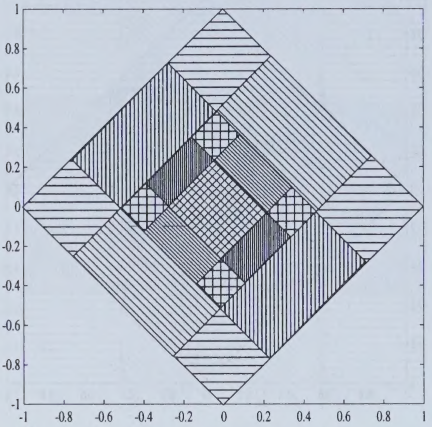


Figure 4: HVS-based wavelet decomposition of frequency space, 3 levels.

4.1 HVS-based Bit Allocation

The allocation of bits to the various subbands of a subband system using knowledge of the HVS is discussed in this section. The 1-D case will be examined first and extension to the 2-D case will be described next.

A two-band subband system with a low-pass channel and a high-pass channel can be cascaded to form a system with several subbands of different bandwidth and sampling rate. Due to the downsampling, the sampling rate of the subband signals will be lower than the original sampling rate by a factor of two for every stage the signal passes through. The bandwidth of the subband signal is also decreased by a factor of two for each stage so the sampling rate is proportional to its bandwidth. The problem of bit allocation is to assign a bit rate in bits per sample to each of the subband signals such that the overall bit rate of the total system, R_T, satisfies an overall bit rate requirement $R_T \leq R_{req}$. The basic idea behind the HVS based bit allocation is that the number of bits allocated to a subband signal should be based on the importance of that signal's information determined by sensitivity of the HVS to that signal from (1) and (2).

The bit allocation for a subband signal w_n will now be explained. Let w_n be band-limited between f_1 and f_2, see Figure 5. The number of bits allocated to this signal is proportional to the area under the log sensitivity function between f_1 and f_2. The number of samples in the signal w_n is proportional to the bandwidth,$2^{-n} f_{max}$, so the bit rate B_n for this subband is described by

$$B_n = \kappa \log S(f_a) \quad \text{bits/sample.} \tag{4}$$

Given the overall bit rate requirement R_{req}, the smallest value of f_{max} is determined which will satisfy $R_T \leq R_{req}$ with R_T given by

$$R_T = 2^{-M} B_L + \sum_{n=1}^{M} 2^{-n} B_n \tag{5}$$

where B_n is the bit rate in the n-th subband signal and B_L is the bit rate for the low-pass signal. Note that the value of f_a depends on f_{max} and increasing f_{max} will decrease the

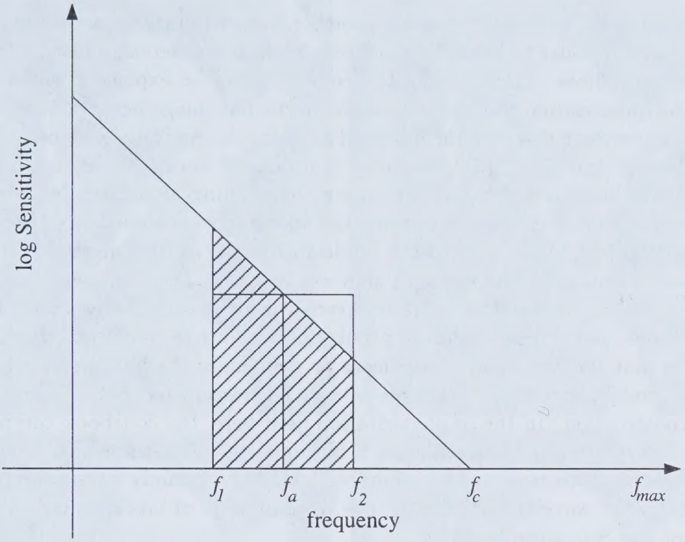


Figure 5: Log sensitivity approximation

value of $\log S(f_a)$. Increasing the value of f_{max} is equivalent to increasing the viewing distance for a particular image. Note that because the number of bits allocated to a subband cannot be negative, $\log S(f_a)$ is taken to be zero for values of f_a which are above f_c, the sensitivity threshold or cutoff frequency. Using this value of f_{max}, the bit rate for each subband is determined by (4) and the overall bit rate will satisfy the given system requirement.

Bit allocation for a 2-D system is carried out in a similar manner. The areas in the 1-D case become volumes and the bandwidths are 2-D. The number of samples is proportional to the area of the bandwidth and the number of bits allocated to the signal is proportional to the volume under the log sensitivity curve for the bandwidth of the subband signal. An equivalent average frequency f_a is determined and either (1) or (2) is used depending on the orientation of f_a.

4.2 Vector Quantization of Wavelet Coefficients

Each of the subband images in the wavelet decomposition are quantized separately. The number of bits allocated to each subband is determined according to the HVS-based method presented in section 4.1. A good discussion of VQ may be found in [14].

Using the VQ method, both the encoder and the decoder must have access to the same codebook. There are two main approaches to codebook design. The first approach uses an ensemble of "typical" images called a training set. The codebook is designed in advance and the fixed codebook needs only to be loaded during VQ of a particular image. This approach, which is used in [8], works quite well if the image being coded is similar to the images in the training set. However, if the image being coded is very different from the images in the trainging set, the results may be quite poor. The second approach produces a codebook which is good for the particular image being coded. For the encoder and decoder to have the same codebook, the codebook must be transmitted along with the indices. Transmitting the codebook can be expensive but it allows for adapting the quantization to the needs of the particular image being coded.

There are two main stages in the design of a codebook. An initial codebook is obtained in the first stage and this initial codebook is improved iteratively in the second stage. The initial codebook is generated here using the pruning algorithm described in [14]. The initial codebook is then improved through an iterative process called the generalized Lloyd algorithm (GLA), also called the Linde-Buzo-Gray (LBG) method [14]. To avoid excessive computation in the event of slow convergence, the GLA was limited to ten iterations in the implementation. The zero vector is given special treatment. It is added to the codebook, never deleted during pruning, and never recomputed during the GLA. This insures that the zero vector remains as an element of the final codebook.

A larger codebook will require more bits to be transmitted but will provide higher quality reconstruction. In the implementation used here, the codebook components are quantized before transmission using an 8 bit uniform quantizer which is sufficient to reduce codebook distortion to an acceptable level but certainly not optimal given the distribution of the wavelet coefficients. The total number of bits required to transmit a codebook of size S is given by

$$B_{codebook} = 8 n_1 n_2 (S - 1) \tag{6}$$

where the block size is $n_1 \times n_2$ so each vector has $n_1 n_2$ components. Note that the zero vector does not need to be transmitted. Improved codebook quantization is an area for future research.

The index coding was modified to take advantage of the frequency of zero vectors. A single bit is transmitted to indicate whether the vector is zero or non-zero. If the vector is non-zero, this bit is followed by an index of size $\lceil \log_2 (S-1) \rceil$ bits into the $S-1$ non-zero vectors. It was found that the percentage of zero vectors ranged from 80 to 96 percent of the total number of vectors for the images investigated. Caution should be exercised in the use of these numbers since they were obtained from a relatively small number of images. The point is that well over half the vectors are the zero vector, justifying the use of one bit to indicate a zero or non-zero vector. If 88 percent of the vectors are the zero vector, the number of bits needed to transmit the indices with this improved coding technique is

$$B_{index} = \frac{N_1 N_2}{n_1 n_2}((.88)1 + (.12)(1 + \lceil \log_2 (S-1) \rceil)) \tag{7}$$

where the image size is $N_1 \times N_2$. It is assumed image size is a multiple of block size.

In the VQ of a particular subimage, it is desirable to have as large a codebook as possible given the allowable bit rate for that subimage. From (6) and (7) the total bit rate for the VQ of a subimage is given by

$$R_{total} = \frac{8 n_1 n_2 (S-1)}{N_1 N_2} + \frac{(1 + (.12) \lceil \log_2 (S-1) \rceil)}{n_1 n_2} \quad \text{bpp.} \tag{8}$$

The codebook size is chosen as the largest value of S for which R_{total} is less than the bit rate allocated to that subband. Note (8) assumes that 88 percent of the vectors will be zero vectors. The actual bit rate attained will be more or less depending on the actual percentage of zero vectors. After VQ encoding and decoding the reconstructed wavelet coefficients have only the largest values preserved while the smaller values are set to zero.

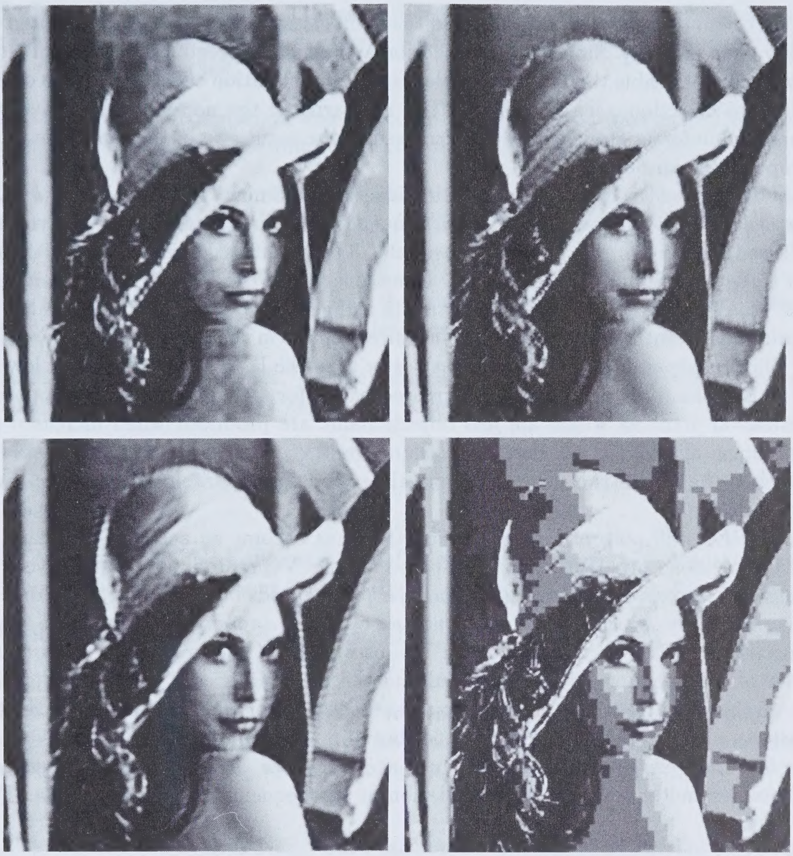


Figure 6: Comparison of reconstructed images: (a) 12-th order separable wavelet VQ, 0.155 bpp; (b) non-separable wavelet VQ, 0.205 bpp; (c) 12-th order HVS-based wavelet VQ, 0.169 bpp (d) JPEG, 0.167 bpp.

4.3 Comparison of the Reconstructed Images

The three wavelet decompositions presented in Section 3 are compared in this section. All of these decompositions give perfect reconstruction if the transform coefficients are not quantized. If the transform coefficients are compressed using the VQ method discussed in section 4.2, artifacts of the quantization can be found in the reconstructed image. These artifacts appear differently in each of the three wavelet decompositions.

The separable wavelet decomposition is the most common 2-D wavelet decomposition. As discussed in Section 3, the rectangular subbands do not match the diamond shaped response of the HVS. Additionally, since the filtering is applied in the horizontal and vertical directions, the quantization artifacts are also aligned horizontally and vertically, see Figure 6a.

The diamond shaped subbands of the non-separable wavelet decomposition are a better fit to the HVS diamond shaped frequency response. Quantization artifacts at one scale are oriented along the diagonals and the artifacts at the next scale are oriented horizontally and vertically. Since the quantization artifacts are spread along more directions, they are less noticeable than the separable wavelet quantization effects which are oriented in only two directions, see Figure 6b. The regularity of the non-separable wavelet decomposition unfortunately cannot be increased as the regularity of the separable wavelet decomposition can be increased.

The HVS-based wavelet decomposition also uses diamond shaped subbands to take advantage of the diamond shaped HVS frequency response. After the first level, all of the filtering is performed along the diagonals. The quantization artifacts are therefore found to be oriented along the diagonal directions where the HVS is least sensitive, see Figure 6c. The orientation of all the artifacts along two directions is less desirable than spreading the quantization artifacts in four directions as in the non-separable decomposition. The non-separable system performs better than the HVS-based system for low filter orders. But the filter order can be increased easily for the HVS-based decomposition. Quantization artifacts which are less objectionable are produced when increased filter order reduces aliasing. For higher filter orders, the HVS-based reconstructed images look better than the reconstructed images from either the non-separable or separable wavelet decompositions.

The JPEG still picture compression method has become an accepted standard. For light compression both JPEG and HVS-based wavelet VQ reproduce the image very well. It is difficult to compare the artifacts from the two methods because they are so different. Quality measures which do not take into account the properties of the HVS are not appropriate. For heavy compression, JPEG loses much of the image information as the number of quantization levels is reduced and the blocking effect takes over, Figure 6d. Most of the high frequency information has been lost in the HVS-based wavelet VQ reconstructed image, shown in Figure 6c, but the low frequency information has been preserved. The image is degraded but not as severely as the JPEG image because the bit allocation method puts more degradation in the frequency regions where the HVS is less sensitive.

5 CONCLUSION

An overview of important properties of the HVS was provided. Existing separable and non-separable wavelet decompositions were discussed and a new HVS-based wavelet decomposition was introduced. A technique for vector quantization of wavelet transform coefficients with bit allocation based on useful properties of the HVS was described. This quantization was used on the three wavelet decompositions under study and the HVS-based wavelet decomposition had the best reconstruction of a compressed image. The HVS-based wavelet decomposition was also found to be superior to JPEG at low bit rates and comparable to JPEG at higher rates.

REFERENCES

[1] G. K. Wallace, "The JPEG still picture compression standard," *Communications of the ACM*, vol. 34, pp. 30–44, Apr. 1991.

[2] G. K. Wallace, "The JPEG picture compression standard," *IEEE Trans. on Consumer Electronics*, vol. 38, pp. 18–34, Feb. 1992.

[3] S. G. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 11, pp. 674–693, July 1989.

[4] I. Daubechies, "Orthonormal bases of compactly supported wavelets," *Communications on Pure and Applied Mathematics*, vol. XLI, pp. 909–996, 1988.

[5] J. Kovačević and M. Vetterli, "Nonseparable multidimensional perfect reconstruction filter banks and wavelet bases for $\mathcal{R}^n$," *IEEE Trans. on Information Theory*, vol. 38, pp. 533–555, Mar. 1992.

[6] R. A. DeVore, B. Jawerth, and B. J. Lucier, "Image compression through wavelet transform coding," *IEEE Trans. on Information Theory*, vol. 38, Mar. 1992.

[7] A. S. Lewis and G. Knowles, "Image compression using the 2-D wavelet transform," *IEEE Trans. on Image Processing*, vol. 1, pp. 244–250, Apr. 1992.

[8] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image coding using wavelet transform," *IEEE Trans. on Image Processing*, vol. 1, pp. 205–220, Apr. 1992.

[9] F. W. Campbell, J. J. Kulikowski, and J. Levinson, "The effect of orientation on the visual resolution of gratings," *Journal of Physiology*, vol. 187, pp. 427–436, 1966.

[10] O. Rioul and M. Vetterli, "Wavelets and signal processing," *IEEE Signal Processing Magazine*, vol. 8, no. 4, pp. 14–38, 1991.

[11] R. J. Safranek and J. P. Johnston, "A perceptually tuned sub-band image coder with image dependent quantization and post-quantization data compression," in *Proc. IEEE ICASSP 89*, vol. 3, (Glasgow, Scotland, May 23-26, 1989), pp. 1945–1948, 1989.

[12] J. W. Woods and S. D. O'Neil, "Subband coding of images," *IEEE Trans. on Acoustics, Speech, and Signal Proccessing*, vol. ASSP-34, pp. 1278–1288, Oct. 1987.

[13] J. W. Woods, ed., *Suband Image Coding*. Boston: Kluwer Academic Publishers, 1991.

[14] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*. Boston: Kluwer Academic Publishers, 1992.

EVALUATION OF A FUNCTION OCCURRING IN MAXIMUM-LIKELIHOOD IMAGE-RESTORATION FOR CCD CAMERA DATA†

Donald L. Snyder[1], *Carl W. Helstrom*[2], *Aaron D. Lanterman*[1], and *Mohammad Faisal*[1]
[1]Washington Univ., St. Louis MO 63130 [2]Univ. of Calif., San Diego CA 92037

Summary

As noted by Snyder, Hammoud, and White [1], the following function occurs in the problem of forming maximum-likelihood estimates of object intensities from data acquired with a charge-coupled-device camera:

$$F(r,\mu,\sigma)=\frac{\Sigma_{n=0}^{\infty}(n/n!)\mu^{n}\exp[-(r-n)^{2}/2\sigma^{2}]}{\Sigma_{n=0}^{\infty}(1/n!)\mu^{n}\exp[-(r-n)^{2}/2\sigma^{2}]}.$$

Here, $r \equiv r(y) \in \mathbf{R}^1$ represents the data value read from pixel y of the CCD camera's pixel-array, σ^2 is the variance of the camera's read-out noise, and the value of $\mu \equiv \mu(y) \in \mathbf{R}^1_+$ for pixel y of the camera's pixel-array is given by

$$\mu(y)=\beta(y)\int_{\mathbf{R}^2} p(y|x)\lambda(x)dx+\mu_0(y),$$

where $\beta(\cdot)$ accounts for nonuniform flat-field effects, $p(\cdot|\cdot)$ is the point-spread function of the camera, $\lambda(\cdot)$ is the object intensity function to be estimated, and $\mu_0(\cdot)$ accounts for background radiation, dark current, and bias. Our interest in this function is motivated by ongoing projects to restore images acquired with the Hubble Space Telescope [1] and in optical-sectioning microscopy [2].

This function $F(\cdot,\cdot,\cdot)$ must be evaluated numerous times when producing maximum-likelihood estimates of $\lambda(\cdot)$ via the expectation-maximization algorithm. In this paper, we describe two approaches for accomplishing such evaluations. The first is an approximation based on a Poisson approximation to Gaussian read-out noise when read-out noise is large, as described and used in [1]. The second is an iterative approach for evaluating $F(\cdot,\cdot,\cdot)$ based upon the saddle-point integration methods introduced by Helstrom [3].

Our preliminary conclusions are: 1, the improvement in the quality of images restored using the saddle-point method are not major when a large number (e.g., 500) of the EM iterations are performed in restoring images as described in [1]; 2, the improvement may be significant for a smaller number (e.g., 50) of EM iterations; and 3, the computation time for restoring images using our DEC/MASPAR-12000 parallel computer, having 4096 processing elements, does not increase significantly with the saddle-point method for 1K-by-1K images.

We are grateful to Dr. R. L. White of the Space Telescope Science Institute for helpful discussions and to the STScI for providing synthetic HST data used in our evaluations.

References

1. D. L. Snyder, A. M. Hammoud, and R. L. White, "Image recovery from data acquired with a charge-coupled-device camera," *J. of the Opt. Soc. of Amer.* A, pp. 1014-1023, May 1993.
2. M. I. Miller and S. Joshi, "Maximum *a posteriori* estimation with Good's roughness for three-dimensional optical-sectioning microscopy," *J. Opt. Soc. of Amer.* A, pp. 1078-1085, May 1993.
3. C. W. Helstrom, "Approximate evaluation of detection probabilities in radar and optical communications," *IEEE Transactions on Aerospace and Electronic Systems*, pp. 630-640, July 1978.

---

† This work was supported by the National Science Foundation under grant number MIP-9101991 and by the Division of Research Resources of the National Institutes of Health under grant RR001380.

Efficient Scan Patterns for Image Decorrelation *

N. D. Memon
Computer Science
Arkansas State Univ.
Jonesboro, Arkansas
AR 72467-0070

K. Sayood
Electrical Engineering
Univ. of Nebraska
Lincoln, Nebraska
NE 68588-0551

S. Magliveras
Computer Science
Univ. of Nebraska
Lincoln, Nebraska
NE 68588-0115

Abstract

In general, lossless image compression schemes implicitly assume that an image is scanned in a particular order. Clearly, depending on the picture, a different scanning pattern may give better compression. In this paper we present some new lossless image compression schemes based on the notion of a prediction tree for an image. A traversal of any prediction tree defines a connected scan of the image. The set of all prediction trees defines a family of connected scans. A minimum entropy prediction tree is then a connected scan from this family, that minimizes the zero-order entropy of intensity differences encountered. We show that the problem of computing a minimum entropy tree is NP-Hard. However, we give efficient heuristics for computing low entropy prediction trees. We identify conditions under which the heuristics would yield an optimal result.

In order to encode an image with respect to a specific scan, one needs to first encode the scan. Due to the large number of prediction trees, the cost of encoding an optimal scan turns out to be prohibitively high. This leads to the need for a trade-off between minimizing the number of bits needed to encode the scan and the entropy of the decorrelated image with respect to the scan. We show that a favorable trade-off can indeed be obtained. We outline a few approaches for which implementations were made. The bit rates achieved compare favorably with the JPEG still compression standard for lossless coding.

1 Introduction

There are two basic components in a lossless image compression technique [13], *decorrelation* and *coding*. In the decorrelation step, redundancies among pixels are reduced, resulting in an image which is generally called the *residual image* or *error image*. The original image can be exactly recovered from the residual image. If decorrelation is effective then the residual image has significantly lower zero-order entropy compared to the

---

***This work was supported in part by the NASA Lewis Research Center and by the Center for Communication and Information Science, University of Nebraska - Lincoln.**

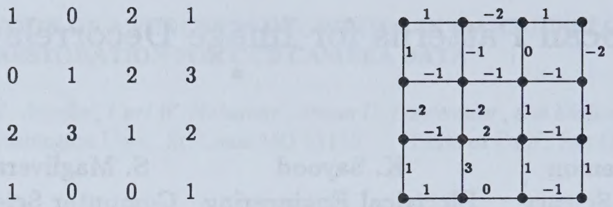


Figure 1: A digital image and its difference graph

original. The efficacy of the decorrelation step is measured by the reduction in zero-order entropy obtained. In this paper we focus on the decorrelation step.

Most image decorrelation schemes reported in literature scan pixels in a fixed order, which does not change from image to image. While spatial causality is not inherent in the image formation process, causality is imposed on an image by processing it in raster scan order. In image compression applications, this practice is historically due to constraints on the number of pixels one could store in main memory. In many applications, however, current technology usually permits us to load the entire image into memory at once. Having the entire two-dimensional structure of the image in memory, one is free to process the pixels in any order one chooses. Clearly, for different images, different scan orders would give better decorrelation.

A natural way to specify an order for scanning an $M \times N$ image, would be to use a permutation of degree $M \times N$. If the pixels are labeled appropriately, then the standard raster scan can be represented by one such permutation, namely, the identity permutation. The question then arises that, given some image decorrelation scheme, what would be the *optimum* way to scan a given image? That is, a scan that maximizes the compression obtained. If the scanning order is to be governed by the two-dimensional structure inherent in the image, then we would expect the scan to be *connected*. By connected we mean that the next pixel visited is always adjacent to some pixel which has already been scanned. If we restrict our attention to connected scans, then for a simple decorrelation scheme, we can provide an answer to the question posed above. The answer leads us to new image dependent decorrelation schemes. The schemes are based on the notion of a prediction tree of an image. Each prediction tree defines a decorrelation scheme. A minimum entropy prediction tree defines a decorrelation scheme that results in a residual image of minimum entropy, over the set of all prediction trees. We show that while the problem of computing a minimum entropy tree is NP-hard, we can obtain efficient heuristics for computing low entropy prediction trees. Due to lack of space, we omit proofs of the results obtained. Detailed proofs can be found in [8].

2 Connected Scans and Prediction trees

Since neighboring pixels in an image are highly correlated, one simple yet surprisingly efficient way to decorrelate an image is to scan the image row by row going from left to right (raster scan order) and replace each pixel value by its difference with respect to the previous pixel. The first (leftmost) pixel in the first (topmost) row is left as is and for the remaining rows, the first pixel is replaced by its difference with respect to the neighboring pixel above. Instead of using a fixed scan, if we are allowed to use any arbitrary scan, what would be the optimal way to scan the image? By optimal we mean a scan that minimizes the entropy of the differences encountered. If we allow the scan to be arbitrary then the scan specified by any permutation that sorts the pixel values would be an optimal scan. If we restrict ourselves to connected scans, however, then the answer is not as simple. In the rest of the section we address this question. We arrive at the answer by a graph theoretic formulation of the problem.

We adopt standard graph theoretic notation and terminology as in [2]. We assume the reader is familiar with the same. Given a digital image P, we construct its *Difference Graph* D by including a vertex for every pixel in P. We then connect vertices representing neighboring pixels with an edge that is weighted with the difference in the intensity values of the corresponding pixels. A digital image and its difference graph are shown in Figure 1. The difference graph shown is obtained by using a *4-neighborhood model.* An *8-neighborhood model* could also be used. It is clear that an image P is completely specified by a single pixel value and any spanning tree T of its difference graph, along with the weights on the edges of T. Given a difference graph D, of a digital image P, we call any spanning tree T, of D a *Prediction Tree* of the image P. Any prediction tree T of an image, yields a connected scan. Given this formulation, the problems related to finding good scans can be abstracted as problems which involve constructing a spanning tree of the difference graph, with desired properties. Hence, the problem of finding a connected scan which minimizes the entropy of the differences encountered is equivalent to constructing a spanning tree T of the difference graph such that the zero-order entropy of the weights $H(T)$ on T is minimized. We call such a tree a *Minimum Entropy Prediction Tree.* Similarly, we can define a *minimum entropy spanning tree* for any weighted graph G. The problem that now arises is that given a weighted graph G, how do we compute a minimal entropy spanning tree of G?

Observe that $H(T)$ depends only on the frequencies of weights on T and not on the specific weights that are involved. A graph G with n vertices has $n-1$ edges in any of its spanning tree T. Hence, T induces a *partition* $\lambda(T) = (\lambda_1\ \lambda_2\ \dots\ \lambda_r)$ of the integer $n-1$, if we simply list the nonzero frequency of occurrence on the weights on T. If any element in the partition λ_j occurs k times then we use the notation λ_j^k to denote this. If we have two partitions $\lambda(T_1)$ and $\lambda(T_2)$ of n, such that $\lambda(T_1)$ is a *refinement* of $\lambda(T_2)$, then we denote this by $\lambda(T_1) \leq \lambda(T_2)$.

Lemma 1 *Let G be a weighted graph and T_1, T_2 be two of its spanning trees. If $\lambda(T_1) \leq \lambda(T_2)$ then $H(T_1) \geq H(T_2)$, with equality if and only if $\lambda(T_1) = \lambda(T_2)$.*

Now we show that given a weighted graph G, computing a minimum entropy spanning tree of G is *NP-Hard* [7]. We do so by reducing the problem of finding an exact three cover by 3-sets (X3C problem) to our minimum entropy problem (ME problem). The X3C problem is known to be *NP-Complete* [7] and can be stated as follows [7]:

Instance: A finite set X with $|X| = 3q$ and a collection C of 3-element subsets of X.
Question: Does C contain an exact cover for X, that is, a sub-collection $C' \subseteq C$ such that every element of X occurs in exactly one member of C'?

An instance I of the X3C problem is said to be a *yes instance* if the answer to the above question is in the affirmative. Otherwise, we call I a *no instance.* Now, given any such $I = (X, C)$ with $X = \{X_1, X_2, \dots, X_{3q}\}$ and $C = \{C_1, C_2, \dots, C_r\}$ we construct a graph G_I as follows:

$$V_{G_I} = \{x_1, x_2, \ldots, x_{3q}\} \cup \{c_1, c_2, \ldots, c_r\}$$

$$E_{G_I} = \{(x_i, c_j) : X_i \in C_j\} \cup \{(c_i, c_j) : j = i + 1\text{and } 1 \leq i < r\}$$

Further, we label the edges of the form (x_i, c_j) with weight j and the edges of the from (c_i, c_j) with weight $r + 1$. Next, we establish the following:

Lemma 2 *Given an instance $I = (X, C)$ of the X3C problem, G_I has a spanning tree T with $\lambda(T) = (3^q\ (r-1)^1)$ if and only if I is a yes instance.*

So we see that only when I is a yes instance, the graph G_I has a spanning tree with a special distribution of weights. Next we show that such a spanning tree has minimum possible entropy, among all spanning trees of G_I.

Lemma 3 *Given an instance I of the X3C problem, let T be any spanning tree of G_I. Then $\lambda(T) \leq (3^q\ (r-1)^1)$.*

Now, we are ready to establish the following:

Proposition 1 *The ME problem is NP-Hard.*

Using the same reduction, the corresponding *decision problem* can be shown to be NP-Complete. So, both the optimization and the decision problems are seen to be hard.

3 Heuristics

Minimum Absolute Weight Heuristic For a typical image the difference between intensity values of two neighboring pixels is low with high probability. Hence, instead of minimizing entropy we could minimize the absolute sum of the weights. The *minimum absolute weight tree (MAW tree)* can be computed in time $O(MN \log\log(MN))$ [3]. In our case, since the graph is *sparse*, a minimum weight spanning tree can be computed in time $O(MN \log^* MN)$ [1] [6], which for all practical purposes amounts to $O(MN)$ ($\log^* 2^{32} = 5$). If we assume that the distribution of prediction errors in an error image is Laplacian with mean zero [12] then minimizing the sum of absolute differences would result in obtaining a tree with minimum entropy. In fact, one can say more.

Lemma 4 *Let T be a minimum entropy tree of a digital image P. Let α_i be the fraction of edges in T with weight i. Suppose that*

$$\alpha_i > \alpha_j \text{ if and only if } |i| < |j|$$

then T is also a minimum absolute weight tree of P.

It should be noted that the above lemma does not guarantee that any minimum absolute weight tree will also be a minimum entropy tree. It merely says that one of the many possible MAW trees is also an ME tree, provided the stipulated conditions are satisfied. However, this fact enables us to construct an efficient algorithm for computing an ME tree, when the condition given in the lemma is satisfied.

---

[1] $\log^* n = \min\{i \geq 0 : \log^i n \leq 1\}$; $\log^i$ is defined by $\log^1 n = \log n$ and $\log^{i+1} n = \log(\log^i n)$

	4-Difference Graph		8-Difference Graph		Best
Image	MAW	MF	MAW	MF	JPEG
USC-Girl	3.90	3.77	3.23	3.02	4.82
Girl	3.04	2.83	2.38	3.02	4.25
Lady	2.80	2.80	2.23	2.23	3.81
House	3.63	3.63	3.13	3.13	4.46
USC-Couple	3.31	3.30	2.83	2.83	4.27
Tree	4.60	4.60	4.01	4.01	5.49
Satellite	5.08	5.08	4.49	4.49	5.89
X-Ray	3.78	3.78	3.18	3.06	5.17

Table 1: Entropy of residual image using prediction trees computed with various heuristics.

Proposition 2 *Let T be a minimum entropy tree of a digital image P. Let α_i be the fraction of edges in T with weight i. Suppose that*

$$\alpha_i > \alpha_j \text{ if and only if } |i| < |j|.$$

Then a minimum entropy tree of P can be computed in time $O(LMN)$.

Maximum Frequency Heuristic Here, we replace the weights in the difference graph by their frequency of occurrence. Then we compute a maximum weight spanning tree of the new graph and the corresponding tree in D is taken as our tree. A tree obtained in this way we call a *Maximum frequency (MF) tree.* In the special case where the frequency of weights decreases with increase in absolute value, this heuristic would be equivalent to a minimum absolute weight heuristic.

Lemma 5 *Let T be a maximum frequency tree of a digital image P. Let α_i be the fraction of edges in T with weight i. Suppose that*

$$\alpha_i > \alpha_j \text{ if and only if } |i| < |j|$$

then T is also a minimum absolute weight tree of P.

The time complexity of this heuristic is again $O(MN)$ due to the sparse nature of our graph. The relationship between ME trees and MF trees is brought out by the following

Proposition 3 *Let D be the difference graph of a digital image P. Let*

$$\bar{\alpha} = (\alpha_0, \alpha_1, \alpha_{-1}, \ldots, \alpha_{L-1}, \alpha_{-L+1})$$

be the vector of frequencies of the weights on D. Let T be an MF tree of P and T' be an ME tree of P. Also let $\bar{\beta} = (\beta_0, \beta_1, \beta_{-1}, \ldots, \beta_{L-1}, \beta_{-L+1})$ be the list of frequencies of the weights on the ME tree T'. If there exists a permutation π which sorts both $\bar{\alpha}$ and $\bar{\beta}$ into descending order, then the MF tree T is also an ME tree.

Experimental Results In Table 1 we give experimental results on implementing the above heuristics for the set of test images. We also provide the entropy of the residual image obtained by using the best JPEG lossless predictor [14]. By best predictor we mean the predictor from the set of 8 predictors specified by the JPEG standard which gives the lowest residual entropy. The best JPEG predictor will vary from image to image.

We have shown that abstracting the scanning problem results in efficient decorrelation schemes. In the rest of the paper we explore the utility of this approach in the development of practical image compression schemes.

4 Identifying a trade-off

Given an image, we can efficiently compute a prediction tree such that the entropy of its weights is significantly lower than the zero-order entropy of the image. Hence, an efficient encoding of the shape of the tree along with a variable rate encoding of its weights given in depth-first order, could provide a compact representation of the image. Unfortunately, the number of spanning trees for an $M \times N$ grid graph has been shown to be [5]

$$4^{(M-1)(N-1)} \prod_{i=1}^{M-1} \prod_{j=1}^{N-1} \left(\sin^2 \tfrac{i\pi}{2M} + \sin^2 \tfrac{j\pi}{2N} \right) \tag{1}$$

Assuming that spanning trees are all equally probable it would require nearly 1.67 bits per pixel to encode the index corresponding to any specific tree for a 256 × 256 image!

When we use a fixed scan, like the raster scan, the cost of encoding the scan is zero, whereas the entropy of weights encountered is high. On the other hand, if we use an optimal scan which has very low entropy, the cost of encoding the scan is prohibitively high. Does there exists a point between these two extreme situations which gives us a minimum total cost? In the rest of this section, we demonstrate that, under reasonable assumptions, the optimal point indeed lies somewhere in between.

It is well known [11, 12] that pixel values in the residual image can be approximated closely by a discrete version of the Laplacian distribution $\mathcal{L}(a, L)$ whose probability mass function can be written as

$$p(i) = \frac{a}{A} e^{-a|i|} \quad \text{for} \quad -L < i < L \tag{2}$$

where A given by

$$A = \sum_{i=-L+1}^{L-1} a e^{-a|i|} = a \cdot \frac{1 + e^{-a} - 2e^{-aL}}{1 - e^{-a}}$$

with entropy

$$\begin{aligned} H(\mathcal{L}(a, L)) &= \sum_{i=-L+1}^{L-1} -p(i) \log p(i) \\ &= \frac{2a}{A} \left(\tfrac{1}{2} - \tfrac{1-e^{-aL}}{1-e^{a}} \right) \log \frac{a}{A} + \frac{2a^2}{A} \cdot \frac{e^{-a} - Le^{-aL} + (L-1)e^{-a(L+1)}}{(1-e^{-a})^2} \log e \end{aligned} \tag{3}$$

To simplify notation we shall use $H(a)$ to denote the entropy of a Laplacian distribution with parameters a and L. We wish to now investigate what happens to the entropy when one moves from an ME tree T to a neighboring tree T' at distance d from T. Let P be a $M \times N$ image and let the variance of the edge weights on a ME tree T of P be $\sigma_T^2 = 2/a_T^2$ and the variance of the weights on the chords of T be $\sigma_C^2 = t \cdot \sigma_T^2$ for some parameter t. Let the weights on the chords of T be realizations of a random variable X with probability density function $f_X(x)$. Similarly let the weights on the edges of T be realizations of the random variable Y with pdf $f_Y(y)$. If T' is any other spanning tree of P then some of its edges (say α of them) will be the edges of T and the others will be the chords of of T. Therefore, the weights on the edges of T' will be realizations of a random variable Z with a mixture distribution

$$f_Z(z) = \alpha f_X(z) + (1-\alpha) f_Y(z) \tag{4}$$

and entropy $H(a_{T'}) =$

$$\frac{2a_{T'}}{A}\left(\tfrac{1}{2} - \tfrac{1-e^{-a_{T'}L}}{1-e^{a}_{T'}}\right)\log\frac{a_{T'}}{A} + \frac{2a_{T'}^2}{A}\cdot\frac{e^{-a_{T'}} - Le^{-a_{T'}L} + (L-1)e^{-a_{T'}(L+1)}}{(1-e^{-a_{T'}})^2}\log e \tag{5}$$

To complete the analysis, we need to estimate the rate at which we would be able to encode an 'approximation' of the MAW tree, such that the reconstructed tree does not differ from the original by more than d edges. The 4-Difference graph of an $M \times N$ image has $M(N-1) + N(M-1)$ edges in it. Any subgraph of it can be encoded as a binary vector of length $M(N-1) + N(M-1)$. So a tree of this graph could be encoded in as many bits. The Hamming distance between binary encodings of two trees which differ by d edges would be $2d$. The minimum rate at which the tree can be encoded is given by the rate distortion function [1, 4] $R_T(D)$ where

$$D = \frac{2d}{MN-1} \tag{6}$$

Unfortunately, rate distortion functions are difficult to compute for most sources. However, if we consider binary encodings of trees to be sequences that are generated by a *Bernoulli Source* [4] with probability of success $p = \frac{MN-1}{M(N-1)+N(M-1)}$ then we should get an upper bound for the rate distortion function of the sequences that encode binary trees. The rate distortion function for a Bernoulli(p) source with Hamming distortion is given by [4]

$$R_B(D) = \begin{cases} H(p) - H(D) & 0 \le D \le \min\{p, 1-p\} \\ 0 & D > \min\{p, 1-p\} \end{cases}$$

So, an estimate of the total cost of using a tree T', which is at distance d from an MAW tree T is

$$C(T') = \frac{M(N-1) + N(M-1)}{MN} \times R_B\left(\tfrac{2d}{MN-1}\right) + H(a_{T'}) \text{ bits per pixel.} \tag{7}$$

a_T and t are determined by the given image. The only parameter over which we have control is d. We would like to know for what value of d the above expression is minimized. The plot of the function $C(\cdot)$ against d, with t parameterized and $a = 0.2$ is presented in Figure 2. These curves, clearly show that the point of minimum cost lies somewhere in the interval $d = [0, \frac{MN-1}{2}]$.

So there is a trade-off between minimizing the cost to encode a scan and minimizing the entropy of the weights on the scan. In the next section, we briefly list some techniques which are successful in striking a favorable trade-off.

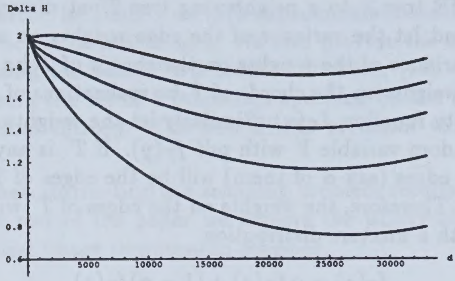


Figure 2: Plot of $C(\cdot)$ against d for $a = 0.20$ and t presented as a parameter. L = M = N = 256.

5 Image Decorrelation Schemes

It is very clear that employing different scanning patterns could potentially lead to significant savings in the number of bits required to represent an image. The zero-order entropy of the weights on the minimum entropy scan is seen to be very low. The problem is to develop algorithms which approximate this scan without paying a heavy price for encoding the scan. Our attempts at this have shown very encouraging results. The bit rates achieved compare favorably with existing techniques for lossless coding. In this section we outline some of the schemes developed and summarize the results obtained. Detailed descriptions of the schemes are presented in [8].

Codebook Techniques There would be many segments within a digital image which possess similar patterns of inter-pixel relationships. If we partition an image into fixed size blocks, it would be reasonable to expect to find scans which are near optimal for a multitude of blocks. Hence, we construct a codebook of scans and for each block in the image identify the best scan from within the codebook. Like any other dictionary or codebook scheme, there are three ways to proceed. We can have a static codebook which we construct from our knowledge about image characteristics in general, or we can have a semiadaptive codebook which is constructed explicitly for each image or class of images. Finally, we could have an adaptive codebook which we construct on the fly as we are encoding the image. In [8], we design algorithms for each of these approaches and give implementation results on a set of test images. Figure 3 compares the performance of codebook techniques. The top most plot drawn with dots and dashes represents the results with a fixed tree (raster scan). The dashed line at the very bottom, represents the entropy of the residual image with local codebooks. The three lines in the middle of these two plots are results obtained with static, semi-adaptive (training set), and backward adaptive codebooks. The lighter shade line represents static codebooks and the darkest shade represents semi-adaptive codebooks. We see that their performance is about the same. Static codebooks perform worse, but still give a 0.25 to 0.50 bit improvement over the raster scan.

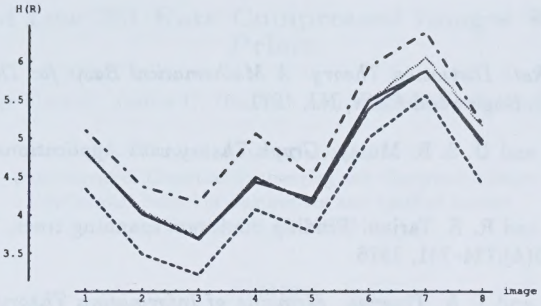


Figure 3: Comparison of codebook techniques.

Lossy Plus Lossless compression In [9] we give a *lossy plus lossless* technique based on prediction trees. It is seen that the prediction tree computed from the lossy image, can be used to for subsequent lossless compression of the original. This gives better performance than the usual practice of transmitting the difference between the lossy image and the original. Improvements of more than 0.5 bits per pixel over the conventional approach were obtained.

Multispectral Image Compression Prediction trees can be used effectively in the situation where we have a sequence of correlated images, like multi-spectral images and certain medical images. In such a situation, the prediction tree obtained from an adjacent image can be used to transmit the current image. Alternatively, a single tree can be constructed that minimizes the absolute sum of prediction errors, over the entire sequence of images. The overhead of transmitting this tree is considerably reduced as the cost is distributed over all the images in the given sequence. Details are again given in [10]. Significant improvements over current techniques were obtained in the case of high spectral resolution image data.

Encoding a Forest of Trees One way to reduce the overhead in coding an optimal prediction tree is to transmit only part of the tree, or select only a forest F that spans only a part of the difference graph, for explicit transmission. The rest of the tree could be filled in by some standard technique. The question that arises is: exactly which subtrees of the difference graph should we transmit? The answer to this and related questions are given in [8].

Encoding a Family of Trees We define a family of trees which can be encoded as binary strings of length $k+\log k$. Given a $k \times k$ image block, the tree from this family that yields a residual block of minimum sum of absolute weights can be computed directly by an 'encoding' algorithm, which outputs a binary string of length $k + \log k$ that encodes this tree. There exists a 'decoding' algorithm that reconstructs the tree from its binary string encoding. Details of the encoding and decoding algorithm are given in [8].

References

[1] T. Berger. *Rate Distrotion Theory: A Mathematical Basis for Data Compression.* Prentice-Hall, Englewood Cilffs, NJ, 1971.

[2] J. A. Bondy and U. S. R. Murty. *Graph Theory with Applications.* North-Holland, 1976.

[3] D. Cheriton and R. E. Tarjan. Finding minimum spanning trees. *SIAM Journal of Computing,* 5(4):724–741, 1976.

[4] T. M. Cover and J. A. Thomas. *Elements of Information Theory.* Wiley Series in Telecommunications. John Wiley and Sons Inc., 1991.

[5] D. Cvetkovic, M. Doob, I. Gutman, and A. Torgasev. *Recent Results in the Theory of Graph Spectra,* volume 36 of *Annals of Discrete Mathematics.* North-Holland, 1988.

[6] M. L. Fredman and R. E. Tarjan. Fibonacci heaps and their uses in improved network optimization algorithms. *Journal of the ACM,* 34(3):596–615, 1987.

[7] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W. H. Freeman and company, 1978.

[8] N. D. Memon. *Image Compression Using Efficient Scan Patterns.* PhD thesis, University of Nebraska - Lincoln, 1992.

[9] N. D. Memon, K. Sayood, and S. Magliveras. A simple technique for enhancing the performance of lossy plus lossless image compression schemes. To appear. *Journal of Electronic Imaging,* July 1993.

[10] N. D. Memon, K. Sayood, and S. S. Magliveras. Lossless compression of multispectral image data. Accepted for publication. *IEEE Transactions on Geosciences and Remote Sensing,* May, 1993.

[11] A. N. Netravili and B. G. Haskell. *Digital Pictures – Representation and Compression.* Applications of Communications Theory Series. Plenum Press, 1988.

[12] Majid Rabbani and Paul W. Jones. *Digital Image Compression Techniques,* volume TT7 of *Tutorial Texts Series.* SPIE Optical Engineering Press, 1991.

[13] S. Todd, G. G. Langdon, and J. J. Rissanen. Parameter reduction and context selection for compression of gray scale images. *IBM J. Res. Develop.,* 29(2):188–193, 1985.

[14] G. K. Wallace. The JPEG still picture compression standard. *Communications of the ACM,* 34(4):31–44, 1991.

Recovery of Low Bit Rate Compressed Images Using Gibbs Priors

Taner Özçelik, James C. Brailean and Aggelos K. Katsaggelos

Northwestern University,
Department of Electrical Engineering and Computer Science
McCormick School of Engineering and Applied Science
Evanston IL, 60208-3118.

Abstract

Most of the image compression techniques produce visible artifacts in the reconstructed image, particularly at low bit rates. In this paper, an iterative MAP estimator for the restoration of the compressed images at low bit rates is proposed. The estimation algorithm makes use of the *a priori* information about the image by assuming a multi-layer Gauss-Markov model. The estimation is performed by the use of a Kalman filter and projection onto the constraint sets. According to this scheme, the unwanted degradations such as the *blocking artifacts* are removed from the observed compressed image, while, through the use of the nonstationary Gauss-Markov model, boundaries in the original image are preserved. Additionally, the *a priori* information about the transform of the original image is incorporated into the estimation process by projecting the image onto the constraint sets at each iteration. The performance of the proposed algorithm was tested on JPEG compressed images. It is shown to be effective in removing the coding artifacts present in the low bit rate compressed images.

1 Introduction

Image data compression is a very important problem for many emerging applications in the fields of visual communication networks. Among the various compression approaches, transform-based methods are the most popular and have found many applications. However, at high compression ratios such techniques often introduce visible artifacts in the reconstructed image. For example, in the case of block encoded images, severe blocking artifacts occur.

In the past, several approaches have been investigated for the reduction of the artifacts in the compressed images In [1] and [2] simple smoothing of the block borders is proposed. In [4], an iterative blocking reduction technique based on the theory of projection onto convex sets is proposed. It employs a global low pass filtering of the image which does not distinguish between stationary and nonstationary parts of it. In [5] the idea of using the theory of Projection Onto Convex Sets (POCS) for image compression applications is introduced. In [6], [7] the deblocking problem is formulated as a regularized recovery problem. Two methods are proposed for its solution, based on the theories of POCS and constrained least squares. However, no spatial adaptivity of the algorithm is considered.

In this paper, we develop a similar technique to the one proposed in [8] for reducing the artifacts present in compressed images. An iterative technique is proposed which consists of a recursive model-based maximum *a posteriori* (MAP) estimator and a projection operator which projects the image onto the given constraint sets. In order to model the image, we propose a multi-layer Gauss-Markov model which is shown to be very powerful in estimating the non-stationarities in intensity (as well as displacement) field restoration problems [9], [10], [11]. The model has two layers in which the upper layer governs the transitions between the sub-models. Based on the image model, the MAP probability distribution function (pdf) is formulated. Maximizing this pdf results in a Kalman type estimator, whose transition matrix is altered depending on the local characteristics of the image. The proposed technique results in an estimator that provides visually very accurate estimates of the intensity field, as is shown experimentally.

2 Formulation of Models

2.1 Compression Model

Several image coding techniques as well as standards have been introduced in the past. In transform-based encoding the image data are first decorrelated and then quantized (in scalar or vector form). Thus, in general, the lossy effects of the compression algorithms originate from the quantization step and can be modeled by

$$y = T^{-1}Q[Tf] = T^{-1}F_t, \tag{1}$$

where f is the original image, T represents a unitary transform, $Q[\cdot]$ is the quantizer, y is the compressed image and F_t denotes the transmitted coefficients. The form of the transform and the quantizer are dependent on the particular compression scheme. For instance, the still image compression standard, JPEG [12], uses the Discrete Cosine Transform (DCT) applied to 8×8 blocks in the image. The transform coefficients are then quantized using a separate uniform quantizer for each coefficient.

Quantization is a many-to-one mapping. Thus, in general, since the only loss of information occur at the quantization step, several images can produce the same compressed image with the same quantizer. However, knowledge of the quantizer limits the number of solutions to a set determined by the step size of the quantizer. Assuming that the quantizer is known at the decoder, one constraint set for the transform coefficients can be determined from the transmitted coefficients F_t. This constraint set is defined by

$$S_Q(F_t) = \{f : Q[Tf] = F_t\} \tag{2}$$

The objective of this work is to obtain an estimate of f that belongs to the set $S_Q(F_t)$ and also satisfies a prior image model. Such a model is presented in the next section. It is in turn used to obtain a MAP estimate of f.

2.2 Image Model

2.2.1 Coupled Gauss-Markov Model

Let $f(\vec{r})$ represent a particular value in the image intensity field, where $\vec{r} = [m, n]^T$, $1 \leq m \leq M$, $1 \leq n \leq N$ denotes the spatial location, and ^{T} the transpose of a vector or matrix. To model $f(\vec{r})$, we propose using a Coupled Gauss-Markov (CGM) model, which is based on a multi-level Markov random field (MRF) approach. The CGM model consists of two layers, a lower layer or line process, which is used to represent the discontinuities of the random field, and an upper layer that describes the observed intensity values. The line process, $l_f(\vec{r})$, is a means of describing the edges contained within an image. The components of the vector $l_f(\vec{r})$ are binary random variables called line elements. More specifically, $l_f(\vec{r})$ is defined as a set of line elements

$$l_f(\vec{r}) = \left\{ \mathrm{l}_{f(m,n)}(i,j) \right\}, \tag{3}$$

where each line element $\mathrm{l}_{f(m,n)}(i,j)$ defines the presence ($\mathrm{l}_{f(m,n)}(i,j) = 1$) or not ($\mathrm{l}_{f(m,n)}(i,j) = 0$) of a boundary between the intensity values $f(m,n)$ and $f(m-i, n-j)$. The CGM model [9], [11] results from the coupling of this line process with several Gauss-Markov (GM) models. It is given by

$$f(\vec{r}) = \sum_{i,j \in \Re\oplus} c_{i,j}^{l_f(\vec{r})} f(m-i, n-j) + v^{l_f(\vec{r})}(\vec{r}), \tag{4}$$

where $c_{i,j}^{l_f(\vec{r})}$ are the prediction coefficients, $v^{l(\vec{r})}(\vec{r})$ is a zero mean Gaussian noise component with variance $\sigma_v^{2\; l_f(\vec{r})}$, and $\Re\oplus$ denotes the region of support. The dependence of $c_{i,j}^{l_f(\vec{r})}$, $v^{l_f(\vec{r})}(\vec{r})$ and $\sigma_v^{2\; l_f(\vec{r})}(\vec{r})$ on $l_f(\vec{r})$ can be explained as follows. With knowledge of $l_f(\vec{r})$, a model is chosen that excludes the intensity values that are separated by a discontinuity. This is accomplished by setting to zero the prediction parameter ($c_{i,j}^{l_f(\vec{r})}$) that corresponds to an activated line element ($\mathrm{l}_{f(m,n)}(i,j) = 1$), which also effects the uncertainty ($v^{l_f(\vec{r})}(\vec{r})$) in the model's accuracy.

A feature of the CGM model is that it is Markovian. It is known from the Gibbs equivalence [3] that a Markov random field with respect to a neighborhood system $\mathcal{N}$ is uniquely characterized by a Gibbs distribution with respect to $\mathcal{N}$. The neighborhood system for the CGM model is the nonsymmetric half plane $\Re\oplus$. Since our ultimate goal is to determine the MAP estimate $\hat{f}(\vec{r})$ from the observed image $y(\vec{r})$, we therefore concentrate on determining the mixed joint probability density function (mjpdf) of f and l_f, conditioned on y. Such a function can be expressed as

$$\rho(f, l_f | y) = \frac{1}{Z} \exp\left\{ -\frac{U_f(f|l_f, y) + U_{l_f}(l_f|y)}{\beta} \right\}, \tag{5}$$

where β is a parameter reflecting the "activity" of the sample field, Z is a normalizing constant independent of f and l_f, and U_f represents the energy function of f. The energy function, $U_f(f|l_f, y)$ is of the form

$$U_f(f|l_f, y) = \sum_{i,j \in \Re\oplus} V_f(f|l_f, y), \tag{6}$$

where $V_f(f|l_f, y)$ is the potential function and the summation is over the set of sites in the neighborhood system surrounding the location (m,n). Clearly, the potential function V_f, and therefore the corresponding energy function U_{l_f} are crucial to the characterization of the properties of the field.

Since each of the submodels are Gaussian, we can obtained the following expression for the potential function $V_f(f|l_f, y)$,

$$V_f(f|l_f, y) = \begin{cases} \dfrac{-\left[2f(m,n) - \sum_{i,j\in\Re} c_{i,j}^{l_f(\vec{r})} f(m-i,n-j)\right] c_{l,k}^{l_f(\vec{r})} f(m-p,n-q)}{2\sigma_v^{2\ l_f(\vec{r})}} & \text{if } (m-p, n-q) \in \Re\oplus \\ \dfrac{f_k{}^2(m,n)}{2\sigma_v^{2\ l_f(\vec{r})}} & \text{if } p = q = 0. \end{cases} \tag{7}$$

The advantage of characterizing the CGM model using this energy function can be explained as follows. If the intensity field begins to deviate from the *a priori* assumptions provided by the GM submodels, then the energy function U_f is increased by the cost V_f, respectively. However, if the line elements are "turned on" ($\mathrm{l}_{f(m,n)}(i,j) = 1$), then there is no cost associated with the corresponding pair of pixels in the field. Therefore, the introduction of abrupt changes in the intensity field is not penalized.

Since we know *a priori* that the intensity field is generally smooth, we want to prevent the line field $l_f(\vec{r})$ from being on everywhere. Therefore, the energy function for each line process is increased for each activated line element. By increasing the energy, we are effectively lowering the probability that a boundary exists in a particular location in the field. Also, we use the energy functions to incorporate additional knowledge we have about the structure of the field. That is, with the knowledge of the states of the past line elements we encourage continuity through the neighborhood $\Re\oplus$, while penalizing crossings, isolated discontinuities, parallel lines, blocking artifacts, and inconsistencies in the line field. The existence of secondary correlation paths around an "activated" line element results in an estimated line field that is inconsistent.

To discourage the above traits, while at the same time encouraging continuity in the line field, we propose the following energy function

$$\begin{aligned} U_{l_f}(l_f|y) &= \lambda_l \sum_{i,j\in\Re\oplus} \mathrm{l}_{f(m,n)}(i,j) + \lambda_e \sum_{i,j\in\Re\oplus} |f(m,n) - f(m-i,n-j)| \left(1 - \mathrm{l}_{f(m,n)}(i,j)\right) \\ &+ \lambda_p \left\{ \sum_{j=\{-1,0\}} \mathrm{l}_{f(m,n-j)}(1,j)\mathrm{l}_{f(m-1,n-j)}(1,j) + \mathrm{l}_{f(m,n)}(0,1)\mathrm{l}_{f(m,n-1)}(0,1) \right\} \\ &+ \lambda_d U_d(\vec{r}|f) + \lambda_i U_i(\vec{r}|f). \end{aligned} \tag{8}$$

The first term of Eq. (8) is the cost associated with the introduction of a line element into the intensity field. More specifically, through this term we take advantage of the knowledge that the intensity field is generally smooth. Therefore a cost is assigned for each activated line element. The second term penalizes if a line element is not activated in areas where abrupt changes occur in the intensity field. The third term prevents the formation of parallel lines. If the previous line elements in the horizontal ($\mathrm{l}_{f(m,n-1)}$), or the vertical ($\mathrm{l}_{f(m-1,n)}$) direction are on, then the probability is small that the adjacent elements should be on. Finally, the last term encourages the formation of continuous contours of discontinuities, and penalizes the formation of isolated line elements and inconsistencies in the line field [11]. That is, through $U_d(\vec{r}|f)$ we encourage the formation of continuous line element contours, while penalizing crosses and isolated points. Through the term $U_i(\vec{r}|f)$, we penalize inconsistent formations of the line process. As was mentioned above, these costs through the neighborhood $\Re\oplus$, taking into account the state of the surrounding line elements. It should also be noted that we take advantage of the available *a priori* knowledge of the block boundary locations, by not allowing a new line element to be introduced when $\vec{r}$ is a multiple of 8 in either the m or n direction. Therefore, in the areas where blocking artifacts are prevalent, the CGM model still accurately characterizes the image intensity field.

2.2.2 Dynamic Model

To develop the proposed estimator, we formulate the CGM model using dynamic equations. This is accomplished by defining a reduced-order state vector [13] for the intensity field,

$$\mathbf{S}_f(\vec{r}) = [f(m,n), f(m,n-1), f(m-1,n+2), f(m-1,n+1), f(m-1,n)]^T . \tag{9}$$

It should be noted that $\mathbf{S}_f(m,n)$ is defined such that $\mathbf{S}_f(m,n-1)$ contains the proper elements needed to satisfy the models' region of support. This allows the field to be expressed as a first order dynamic system, according to

$$\mathbf{S}_f(\vec{r}) = \mathcal{C}^{l_f(\vec{r})}\mathbf{S}_f(m,n-1) + \mathcal{B}_f\mathbf{U}_f(\vec{r}) + \mathbf{W}^{l_f(\vec{r})}(\vec{r}), \tag{10}$$

where $\mathcal{C}^{l_f(\vec{r})}$ is the matrix of the corresponding prediction coefficients from Eqs. (4). $\mathbf{U}_f(\vec{r})$ is defined as

$$\mathbf{U}_f(\vec{r}) = \begin{bmatrix} 0 \\ f(m-1,n+2) \end{bmatrix}, \tag{11}$$

The remaining matrices are defined as follows,

$$\begin{aligned} \mathcal{B}_f &= \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}^T \\ \mathbf{W}_f^{l_f(\vec{r})}(\vec{r}) &= \begin{bmatrix} v^{l_f(\vec{r})}(\vec{r}) & 0 & 0 & 0 & 0 \end{bmatrix}^T, \end{aligned}$$

where

$$\mathcal{E}\left\{\mathbf{W}_f^{l_f(\vec{r})}\mathbf{W}_f^{T\, l_f(\vec{r})}\right\} = \mathbf{R}_{\mathbf{W}_f}, \tag{12}$$

and $\mathcal{E}\left\{\cdot\right\}$ denotes expectation.

The mixed joint *a posteriori* pdf of the intensity and the line fields, can now be expressed in terms of $\mathbf{S}_f(\vec{r})$and $l_f(\vec{r})$ as

$$\begin{aligned} \rho(\mathbf{S}_f(\vec{r}), l_f(\vec{r})|y) &= \rho(y|\mathbf{S}_f(\vec{r}), l_f(\vec{r}))\rho(\mathbf{S}_f(\vec{r}), l_f(\vec{r})) \\ &= \rho(y|\mathbf{S}_f(\vec{r}), l_f(\vec{r}))\rho(\mathbf{S}_f(\vec{r})|l_f(\vec{r}))P(l_f(\vec{r})), \end{aligned} \tag{13}$$

which is the form of the *a posteriori* pdf used in the development of the proposed estimator. The division by $\rho(y)$ is not shown in Eq. (13) since it is independent of the $\mathbf{S}_f(\vec{r})$ and $l_f(\vec{r})$.

3 Iterative MAP Estimator

The proposed estimator has two steps. First, a MAP estimate of the image based on Eq. (13) is obtained by the use of the CGM model described in the previous section. Second, the output of the MAP estimator is projected onto the constraint set given by Eq. (2). This process is performed iteratively until an acceptable quality of the reconstructed image is achieved.

To estimate the original image from the compressed image, we maximize the mixed joint *a posteriori* pdf with respect to $f(\vec{r})$ and the corresponding line process $l_f(\vec{r})$. We begin the maximization process by assuming that the line process is known *a priori*, that is $\mathbf{P}(l_f(\vec{r})) = 1$. Therefore, the objective is now to find $\hat{\mathbf{S}}_f(\vec{r})$ such that $\rho(\mathbf{S}_f(\vec{r})|l_f(\vec{r}))$ is maximized. Since the conditional density $\rho(y|\mathbf{S}_f, l_f)$ is uniform, it is ignored during the maximization process. We know from Eq. (4) that the conditional density $\rho(\mathbf{S}_f|l_f)$ is Gaussian. Therefore, if we assume that the line process is known, then the resulting density is clearly Gaussian. Furthermore, the recursive MAP estimator has the form of a Kalman filter [14].

$$\mathbf{S}_f^-(\vec{r}) = \mathbf{A}^{l_f(\vec{r})}\mathbf{S}_f^+(m, n-1) \tag{14}$$

$$\mathbf{P}_f^-(\vec{r}) = \mathbf{A}^{l_f(\vec{r})}\mathbf{P}_f^+(m, n-1)\mathbf{A}^{Tl_f(\vec{r})} + \mathbf{Z}\mathbf{R}_{W_f^{l_f(\vec{r})}}\mathbf{Z}^T \tag{15}$$

$$\mathbf{S}_f^+(\vec{r}) = \mathbf{S}_f^-(\vec{r}) + \mathbf{K}(\vec{r})[y(\vec{r}) - \mathbf{H}\mathbf{S}_f^-(\vec{r})] \tag{16}$$

$$\mathbf{K}(\vec{r}) = \mathbf{P}_f^-(\vec{r})\mathbf{H}[\mathbf{H}^T\mathbf{P}_f^-(\vec{r})\mathbf{H} + \Gamma]^{-1} \tag{17}$$

$$\mathbf{P}_f^+(\vec{r}) = [\mathbf{I} - \mathbf{K}(\vec{r})\mathbf{H}]\mathbf{P}_f^-(\vec{r}) \tag{18}$$

Γ represents a measure of the global degradation caused by the quantization process and $\mathbf{Z}$ and $\mathbf{H}$ are as shown below:

$$\mathbf{Z} = \begin{bmatrix} \mathbf{I}\,0\,0\,0\,0 \\ 0\,0\,\mathbf{I}\,0\,0 \end{bmatrix}^T \tag{19}$$

and $\mathbf{H} = [\mathbf{I}, 0, 0, 0, 0]^T$.

The above intensity field estimation algorithm is dependent upon the line process. Thus the estimate of the intensity field is determined by deciding which configuration of $l_f(\vec{r})$ and the resulting $\hat{\mathbf{S}}_f^+(\vec{r})$ maximize Eq. (13). Specifically, the estimation algorithm can be described as L extended Kalman filters operating in parallel, where each uses a model corresponding to a different configuration of $l_f(\vec{r})$. The set of line configurations providing the best estimate of $\hat{\mathbf{S}}_f^+$, as determined by a MAP decision criterion, are retained as the estimate $\hat{l}_f(\vec{r})$. A similar maximum likelihood decision criterion was proposed in [15] for the restoration of degraded images.

In summary, we see that the proposed estimation algorithm is capable of adapting to the discontinuities present in the image. The updated intensity signal $\hat{f}^+(\vec{r})$ obtained from Eqs. (14)-(18) has to satisfy the constraints mentioned in the previous section. Therefore the resulting signal $\hat{f}^+(\vec{r})$ is projected into the given constraint sets at each step of the iterative algorithm.

4 Experimental Results

In this section, we present experimental results that illustrate the effectiveness of the restoration algorithm. In the specific implementation of the proposed restoration algorithm $\Re_{\oplus}$ is chosen to be a non-symmetric half plane support region to allow for recursive computability. Figure 1 shows the "lena" image after it has been compressed by JPEG with a compression ratio of 30:1. At this compression ratio the *blocking artifacts* are very noticeable. Figure 2 shows the result of the proposed restoration algorithm after 3 iterations. The number of iterations depends upon the severity of the coding artifacts present in the observed image. Figures 3 and 4 show the enlarged portions of Figs. 1 and 2 respectively. It is shown that the proposed algorithm removes the unwanted degradations such as the *blocking artifacts* very effectively while the discontinuities in the original image are preserved. Therefore the proposed restoration algorithm can be used to achieve higher compression ratios. Additionally, the proposed algorithm does not require any additional information rather than the compressed image itself.

5 Conclusion

In this paper, an iterative MAP estimator for the restoration of the compressed images at low bit rates is proposed. The estimation algorithm makes use of the *a priori* information about the image by assuming a multi-layer Gauss-Markov model. The estimation is performed by the use of an extended Kalman filter and projection onto the constraint sets. The advantages of the proposed restoration scheme are threefold. First, the unwanted degradations such as the *blocking artifacts* are removed from the observed compressed image. Second, through the use of the nonstationary CGM model, boundaries in the original image are preserved. Third, the *a priori* information about the transform of the original image is incorporated into the estimation process by projecting the image onto the constraint sets at each iteration. The performance of the proposed algorithm was tested on standard images and shown to be effective in removing the coding artifacts present in the low bit rate compressed images.

References

[1] H. C. Reeves and J. S. Lim, "Reduction of blocking effects in image coding," *Optical Eng.*, Vol. 23, No. 1, pp.34-37, Jan/Feb 1984.

[2] B. Ramamurthi and A. Gersho, "Nonlinear Space Variant Post-processing of Block coded Images," *IEEE Trans. on Acoust., Speech and Signal Proc.*, vol. ASSP-34, no.5, pp. 1258-1268, October 1986.

[3] S. Geman and D. Geman, "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images" *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-6, no. 6, pp. 721-741, Nov. 1984.

[4] R. Rosenholtz and A. Zakhor, "Iterative Procedures for Reduction of Blocking Effects in Transform Image Coding", *IEEE Trans. on Circuits and Syst. for Video Tech.*, vol.2, no.1, pp. 91-95, March 1992.

[5] : P. Santago and S. A. Rajala, "Using convex set techniques for combined pixel and frequency domain coding of time-varying images," *IEEE Journal on Selected Areas in Communications*, Vol. 5, No. 7, pp. 1127-1139, August 1987.

[6] Y. Yang, N.P. Galatsanos and A.K. Katsaggelos, "Iterative Projection Algorithms for Removing the Blocking Artifacts of Block-DCT Compressed Images," *Proc. 1993 Int. Conf. Acoust., Speech, Signal Processing*, pp. V-405-408, Minneapolis, MN, April 1993.

[7] Y. Yang, N. P. Galatsanos and A. K. Katsaggelos, "Regularized Image Reconstruction from Incomplete Block Discrete Cosine Transform Data",*to appear, IEEE Trans. on Circuits and Systems for Video Technology*, October 1993.

[8] R. L. Stevenson, "Reduction of Coding Artifacts in Transform Image Coding," IEEE ICASSP '93, Minneapolis, pp. V-401–V-404, April 1993.

[9] J.C. Brailean and A.K. Katsaggelos, "Recursive displacement estimation and restoration of noisy-blurred image sequences," *Proc. IEEE ICASSP '93*, Minneapolis, pp. V-273-276, April 1993.

[10] J.C. Brailean and A.K. Katsaggelos, "Displacement field estimation in noisy image sequences," *Proc. EUSIPCO-92*, pp. 1319-1322, Aug., 1992.

[11] J.C. Brailean, *Recursive displacement field estimation with application to image sequence processing*, Ph.D Dissertation, Dept. of EECS, Northwestern University, Evanston, IL, August 1993.

[12] G. K. Wallace, "The JPEG Picture Compression Standard," *IEEE Trans. on Consumer Electronics*, vol. 38, no. 1, pp. 18-34, February 1992.

[13] D. L. Angwin and H. Kaufman, "Image restoration using reduced order models," *Signal Processing*, vol. 16, pp. 21-28, Dec 1988.

[14] P. S. Maybeck *Stochastic Models, Estimation, and Control*. vol. 2, New York: Academic Press, 1982.

[15] A.M. Tekalp, H. Kaufman, and J.W. Woods "Edge-adaptive Kalman filtering for image restoration with ringing suppression," *IEEE Trans. Acoust, Speech, and Image Proc.*, vol. 37, no. 6, pp. 892-899, June 1989.

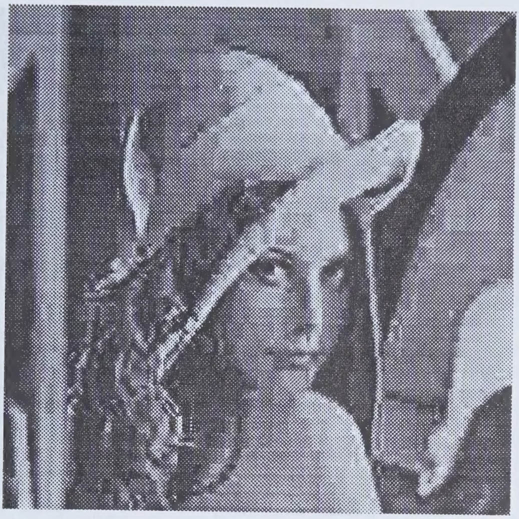


Figure 1: Compressed "*Lena*" (Compression Ratio: ≈ 30:1).

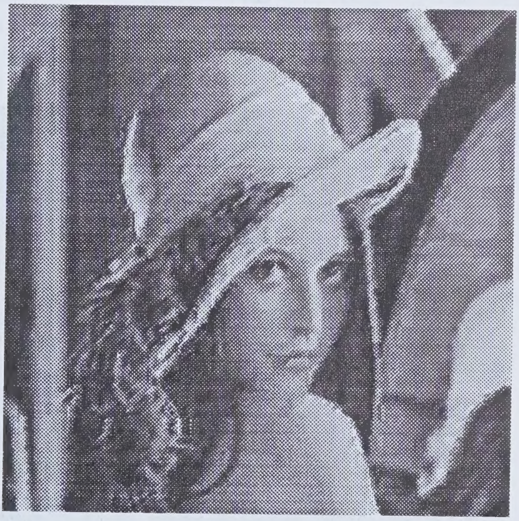


Figure 2: Restored "*Lena*" by the proposed algorithm (Compression Ratio: ≈ 30:1).

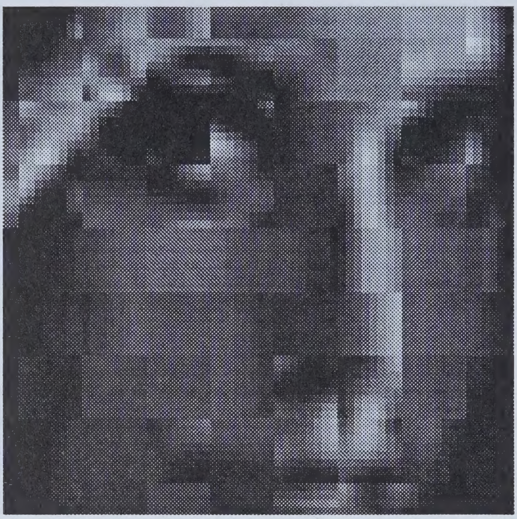


Figure 3: Compressed enlarged "*Lena*" (Compression Ratio: ≈ 30:1).

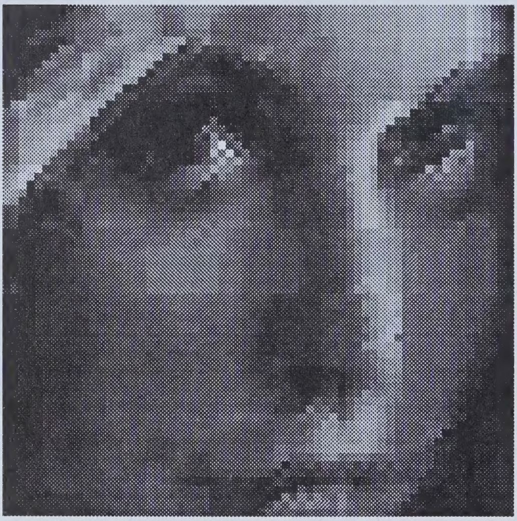


Figure 4: Restored enlarged "*Lena*" by the proposed algorithm (Compression Ratio: ≈ 30:1).

Block Processing - An Unsupervised Non-model-based Image Analysis Approach

Tianhu Lei, Zuo Zhao, and Wilfred Sewchand
University of Maryland, Baltimore

Most of image analysis operations are stochastic model-based approaches. At the lower-level analysis, the observed image is modeled by the random fields, the fractals, and the wavelets, etc [1]. The image analysis (unsupervised) is decomposed to and formulated as a detection (of the number of image primitives), an estimation (of the parameters of the model), and a classification (of the pixel into the destinctive primitives) problem. At the higher-level analysis, the interpretation labels are considered as a symbolic random field to account for the uncertainty in assigning object labels to the image primitives. The image analysis is formulated as an estimation problem, such as MAP, etc. Stochastic model-based approaches demonstrate the great success in the various fields of image analysis. The shortcomings and limitations, however, do exist. It is impossible to create an universal model to represent the various types of images. Even for a specified image, it is difficult to validate the conditions for the model to fit the image. Thus, the heuristic assumptions are always imposed.

This study is confined in the computed image and the lower-level analysis. The examples of the computed image may be x-ray CT, MRI, etc. The basic common feature of the computed image is that its image reconstruction is actually a transform from a random process (a set of random variables - physical measurements) to another random process (an another set of random variables - image pixels). Our previous studies derived several statistical properties about image pixels of the computed image [2]. Based these properties and using some array signal processing techniques [3], the Block Processing - an unsupervised non-model-based image analysis approach - is developed. Its first part - image detection - is reported in this paper.

For the simplicity, consider a $N \times N$ computed image, here $N = M \times L$ and $M > L^2$, M, N, and L are integers. By sampling this image with the step size M in both horizontal and vertical directions, the L^2 sampled sub-images are created. Each of these L^2 sub-images has $M \times M$ pixels and is refered to as a Block Processor. Let x_{lm} $(1 \leq l \leq L^2,\ 1 \leq m \leq M^2)$ denote the value of the $m-th$ pixel in the $l-th$ Block Processor. Define a L^2-dimensional vector, Y,

$$Y = (x_1, \cdots, x_l, \cdots, x_{L^2})^T \tag{1}$$

and its $m-th$ sample vector, Y_m,

$$Y_m = (x_{1m}, \cdots, x_{lm}, \cdots, x_{L^2m})^T \tag{2}$$

The sample convariance matrix $\bar{\Sigma}$ of Y is

$$\bar{\Sigma} = \frac{1}{M^2} \sum_{m=1}^{M^2} (Y_m - \bar{Y})(Y_m - \bar{Y})^T \tag{3}$$

where $\bar{Y}$ is the sample mean vector of Y.

Assume that the image contains K_1 regions and these K_1 regions belong to K types, here $K \leq K_1$ and $K < L^2$, K_1 and K are unknown and to be determined. Compute the eigenvalues of the sample convariance matrix $\bar{\Sigma}$, λ_i, $(1 \leq i \leq L^2)$, $\lambda_1 > \lambda_2 > \cdots > \lambda_{L^2}$. Then, substitute λ_i into the Minimum Description Length (MDL) formula [4]

$$MDL(k) = -\log\left(\frac{\prod_{i=k+1}^{L^2} \lambda_i^{1/(L^2-k)}}{\frac{1}{L^2-k}\sum_{i=k+1}^{L^2} \lambda_i}\right)^{(L^2-k)M^2} + \frac{1}{2}k(2L^2-k)\log M^2 \qquad (4)$$

we will have

$$K = Argmin_{1 \leq k \leq L^2}\{MDL(k)\} \qquad (5)$$

After K is determined by the above procedures, the conventional image analysis technique [5] (at the lower-level) can be applied. For instance, Cluster-seeking algorithm and Bayesian criterion will be utilized to estimate the region-dependent parameters and classify the pixels into the regions, respectively. This will complete the lower-level analysis[1].

Two 64×64 images with MRF as the underlying field are created [6]. Gaussian, White noise with zero mean and adjustable variances are added. These two images consist of 4 types of regions. Applying the Block Processor and Block Processing to these two images, we have the results $K = 4$.

The conventional (unsupervised) model-based image analysis approaches requires the iterative, back-forth procedure between detection and estimation. In the case of Block Processing, once the covariance matrix of the observation vectors is formed, since the analytic formula (in the closed form) is available, the number of regions can be straightforward determined. Thus, it does not require the iterative algorithms, and therefore, it needs much less computation time than the former. Clearly, the application of Block Processing in image analysis is the object detection.

REFERENCES

1. A. K. Jain, *Fundamentals of Digital Image Processing*, Prentice Hall Inc., Englewood Cliffs, NJ. 1989
2. T. Lei and W. Sewchand, 'Statistical approach to x-ray CT imaging and its applications in image analysis - Part 1: Statistical analysis of x-ray CT imaging,' *IEEE Trans. on Medical Imaging,* vol. MI-11, no. 1, 1992.
3. M. Wax and T. Kailath, "Detection of signals by information theoretic criteria", *IEEE Trans. on Acoust., Speech, and Signal Proc.*, vol. ASSP-33, Apr. 1985.
4. J.Rissanen, 'A universal prior for integer s and estimation by minimum description length,' *Ann. Stat.,* vol. 11, no. 2, 1983.
5. T. Lei and W. Sewchand, 'Statistical approach to x-ray CT imaging and its applications in image analysis - Part 2: A new stochastic model based image segmentation technique for x-ray CT image,' *IEEE Trans. on Medical Imaging,* vol. MI-11, no. 1, 1992.
6. Y.Q. Yin and P.R. Krishnaiah, 'Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images,' *IEEE Trans. on Pattern Analysis and Machine Intelligence,* vol. PAMI-6, No. 6, Nov. 1984.

---

[1] Due to the page limitation, the validation of the use of array signal processing technique to image analysis is omitted in this short paper.

Parameter Estimation for the Curve Recovery Problem[†]

Barbara E. Schmitz and Robert L. Stevenson
Laboratory for Image and Signal Analysis
Department of Electrical Engineering
The University of Notre Dame
Notre Dame, Indiana 46556

Abstract

Reconstructing curves from noisy, sparse data is an important problem in many areas of image analysis and signal processing. Three techniques for reconstructing curves while preserving discontinuities will be discussed along with the more classical cubic splines fit technique. Each of these techniques has associated with it, one or two parameters which can take on many values. These parameters must be accurately estimated in order to produce the optimal reconstruction of the data. A parameter estimation algorithm will be developed and tested on synthetic data as well as real data.

1. Introduction

In many applications, techniques exist that are capable of forming estimates of a curve's value at sparse locations along the curve. From this sparse information, it is often desired to reconstruct a dense approximation to the curve. In many cases, the curve to be estimated contains discontinuities in both the curve and in the first derivative of the curve. Classical techniques for reconstructing curves, such as the cubic splines technique, tend to smooth over discontinuities present in the curve. Discontinuity preserving curve recovery techniques exist, however these methods introduce an additional parameter which need to be estimated for an accurate reconstruction.

In this paper, the curve reconstruction problem is reviewed and several curve reconstruction techniques are discussed. Associated with these various techniques are parameters which can take on many values. These parameters must be accurately estimated in order to produce the optimal reconstruction of the data. An algorithm will be developed for estimating these parameters and then tested on synthetic data as well as real data.

2. Forming a well-posed problem statement

This section presents a mathematical statement of the curve reconstruction problem and presents the method that will be utilized to find an unique and stable solution. Let a curve in 2-dimensional space be represented explicitly by $y(x)$, where $x \in U \subset \mathbb{R}$, and $y(x)$ is in the functional space V.

2.1. Ill-posed problem statement

A collection of noise corrupted curve location estimates, $\mathcal{S} = \{c_i \in \mathbb{R}\}_{i=1}^{M}$, are observed. The reconstructed curve should pass through or near these constraint locations. The process of constraint formation can be modeled by

$$c_i = y(x_i) + e_i, \qquad i = 1, \dots, M, \tag{1}$$

where $x_i \in U$ are constants, and e_i model the associated measurement errors. The curve reconstruction problem, simply stated, is to form an estimate, $\hat{y}(x)$, of the curve $y(x)$ for all $x \in U$, given the finite collection of noisy and sparse constraints, $\mathcal{S}$. This problem as stated is ill-posed since the sparse and noisy set of estimates only constrain the shape of the curve locally; thus an infinite number of feasible curves will fit the estimates.

[†] This work was supported by a fellowship from the Clare Boothe Luce Foundation and a grant from Apple Computer.

2.2. Solution by Tikhonov regularization

Since the curve reconstruction problem is an ill-posed inverse problem, to obtain a unique and stable reconstruction of the curve from the constraints, supplementary information must be added to make the problem well-posed[1]. The basic principle common to all methods is to use *a priori* knowledge about the properties of the reconstructed curve to resolve conflicts in the estimates, and to restrict the space of possible curves so that the constraints, $\mathcal{S}$, uniquely determine a stable estimate, $\hat{y}(x)$, of the curve. A technique which is often used to form a well-posed problem for many ill-posed inverse problems is the method of Tikhonov regularization[1].

To make the problem well-posed, a continuous operator, known as a regularizing operator, is defined which approximates the inverse operator. To construct a regularizing operator, $R(\cdot,\cdot)$, Tikhonov and Arsenin[1] introduce a stablizing functional, $\Omega[\cdot] : V_1 \rightarrow \mathbb{R}$, with the following properties: (i) the functional $\Omega[\cdot]$ is a continuous nonnegative functional defined on a subset V_1 of V that is everywhere dense in V, (ii) the curve $\hat{y}(x)$ belongs to the domain of definition for $\Omega[\cdot]$, and (iii) for every positive number d, the set of possible reconstructions $y^*(x)$ of V_1, for which $\Omega[y^*(x)] \leq d$, is a compact subset of V_1. This stablizing functional then is used to define the Tikhonov regularizing functional

$$M^\lambda[y^*(x), \mathcal{S}] = \Omega[y^*(x)] + \lambda \sum_{c_i \in \mathcal{S}} \|y^*(x_i) - c_i\|^2, \tag{2}$$

where $\|\cdot\|$ is a norm defined on the space $\mathbb{R}$. This norm is a measure of the distance between a curve and the constraint locations. This term will be large for curves which do not pass near the constraint points. Then, for certain values of λ, the minimization of this functional over V_1 is a regularizing operator,

$$y_\lambda(x) = R(\mathcal{S}, \lambda), \tag{3}$$

where the curve $y_\lambda(x)$ is such that

$$M^\lambda[y_\lambda(x), \mathcal{S}] = \inf_{y^* \in V_1} M^\lambda[y^*(x), \mathcal{S}], \tag{4}$$

and the operator, $R(\cdot,\cdot)$ which approximates the inverse operator represents the operation of finding the curve $y^*(x)$ which minimizes (2) for a given λ and set of constraints $\{c_i\}$. This regularized solution, $y_\lambda(x)$ will be used as the curve estimate, $\hat{y}(x)$. Note that the regularizing operator for a particular problem is in general not unique; there may exist many operators which stabilize the ill-posed problem. The choice of the particular operator and the value of the regularization parameter λ is based on supplementary information pertaining to the problem.

3. Curve reconstruction

In this section a Tikhonov regularization functional is defined for the problem of reconstructing curves. How the stabilizing functional of the regularization functional is defined, leads to different methods of reconstructing curves and thus to estimated curves with different characteristics. The popular method of cubic splines is first described, then we turn to examine currently existing stabilizers which are more robust with respect to discontinuities in the curves than the popular cubic splines method.

3.1. Cubic splines

For cubic splines the smoothness of a curve is measured by integrating the square of the second derivative over the range of the curve. The explicit form of this stablizer designed to reconstruct smooth curves is

$$\Omega[y(x)] = \int_U y''(x)^2 \, dx. \tag{5}$$

This functional has been widely used for smoothing data[2].

Notice that the functional (2) has two terms. To make the reconstruction more robust to outliers in the data (i.e. bad data points) the second term which measures the error between the reconstruction and the data can be modified. Choosing an appropriate value for this parameter will be discussed in the following section. To make the reconstruction more robust to outliers in the smoothness assumption however, the stablizing functional needs to be modified.

3.2. Discontinuity preserving signal models

A method for incorporating the possibility of discontinuities is to incorporate into the stablizer the fact that some discontinuities are consistent with our *a priori* idea of a good reconstruction. For this paper, we will concentrate on stabilizers of the form

$$\Omega[y(x)] = \int \rho[y''(x)]\, dx, \tag{6}$$

where some yet unspecified scalar function, $\rho(\cdot) : \mathbb{R} \to \mathbb{R}$, is used. How the function $\rho(\cdot)$ is defined, in turn leads to different signal models. Note that for the cubic splines fitting,

$$\rho(u) = u^2. \tag{7}$$

One implementation of $\rho(\cdot)$, called the neighborhood interaction function, sets some threshold T for which the weight on $y''(x)$ remains constant for values above the threshold, i.e. define

$$\rho_1(u) = \begin{cases} u^2, & u \le T, \\ T^2, & u > T. \end{cases} \tag{8}$$

This idea, although not derived in this fashion, was used by Blake and Zisserman[3]. In this approach, the function $\rho(\cdot)$ is viewed as a 'smoothness term' which controls the amount of smoothness that should be present at a particular point in the reconstruction. Once a discontinuity is encountered, this smoothness term should in effect be 'turned off' to allow for a steep discontinuity. This thought is implemented by not increasing the cost of differences above a set threshold.

Another implementation, is the generalized absolute function,

$$\rho_2(u) = |u|^P, \tag{9}$$

which was introduced by Bouman and Sauer [4]. The value of the exponent paramenter, P, can be defined as any value between 1 and 2. This value controls the cost associated with sharp discontinuities. Notice that when $P = 1$, the generalized absolute function becomes the function $|u|$ and the reconstruction is not always adequately smoothed. The other extreme, when $P = 2$, results in a cubic splines reconstruction where the discontinuities are smoothed and are not reconstructed. The value of P should be set to a value between these extremes to find an appropriate compromise.

The last function discussed in this paper is to use the Huber minimax function for $\rho(\cdot)$ [5, 6, 7]. Then $\rho(\cdot)$ is of the form

$$\rho_3(u) = \begin{cases} u^2, & u \le T, \\ T^2 + 2T|u - T|, & u > T. \end{cases} \tag{10}$$

This function varies as a square for values below the threshold while varying linearly above the threshold. This enables the reconstruction of smooth parts of the curve as well as allowing for discontinuities to be maintained.

4. Parameter selection

In the reconstruction process discussed in the previous section, there were three parameters which were not defined: the value of the regularizing parameter λ introduced by Tikhonov regularization, the exponent parameter P introduced in the generalized absolute function, and the threshold parameter T introduced in the neighborhood interaction function and the Huber minimax function. In this section, the effect of these parameters and a method for appropriately estimating them will be examined and tested.

4.1. Parameter selection algorithm

The first term of the Tikhonov regularizing functional,

$$M^{\lambda}[y(x), \mathcal{S}] = \Omega[y(x)] + \lambda \sum_{c_i \in \mathcal{S}} ||y(x_i) - c_i||^2, \tag{11}$$

the stabilizing functional $\Omega[\cdot]$, encorporates the *a priori* knowledge of the curve shape. The functions introduced earlier (8) (9) (10) each define $\rho(\cdot)$ differently. Within these functions, the parameters T and P have yet to be defined. Since these variations of the cubic splines function (7) were introduced for the specific purpose of maintaining discontinuities in the reconstructions, these will be called the discontinuity preserving parameters.

The curve and corresponding noisy sparse constraint points shown in Figure 1 will be used as the original signal for the initial parameter investigations of this subsection. Notice that there is a discontinuity in the curve near point 20, and there is a discontinuity in the first derivative of the curve near point 40. Figure 2a shows various reconstructions using the Huber minimax function for different values of the threshold parameter T. Notice that for very small values of T the curves reconstruct in linear segments. As T approaches infinity, the reconstructions again approaches the quadratic case. This behavior is similar to that exhibited by the other discontinuity preserving parameters.

The regularizing parameter, λ, found in the Tikhonov regularizing functional, controls how closely the reconstruction fits to the data. Recall that the minimization of the Tikhonov regularizing functional is the act of finding a compromise between relying on the *a priori* assumptions inherent in $\Omega[\cdot]$, and relying on what the constraint points indicate the reconstruction should look like. The parameter λ is a weighting of how important it is for the curve to be reconstructed near or through the constraint points. Figure 2b shows two reconstructions for different values of λ using the cubic splines function. Notice how the reconstruction is pulled closer to the constraint points when λ is very large, and is allowed more freedom for a smaller value of λ.

As can be seen in Figures 2a and 2b, the reconstruction of a given set of constraints varies significantly depending upon the parameters selected. In order to more accurately reconstruct the data, appropriate parameters must be selected. The values of the parameters which are most appropriate are defined as the value of the parameters associated with the best possible reconstruction for each signal model according to some chosen criterion. That is, the best possible reconstruction using the neighborhood interaction function, will reconstruct the signal by defining the parameters T and λ to be certain values. These values assigned to T and λ are defined as the optimal parameters for the neighborhood interaction function and the chosen criterion.

Assuming the original signal is unknown at the time of parameter selection, a method of determining appropriate values for the parameters can be found by considering the influence of each parameter on the final reconstruction [7].

The parameters T and P are the discontinuity preserving parameters. These parameters control how the curve reconstructs at the points of discontinuities. As can be seen in Figure 2a, for small values of the parameters the curve reconstructs with straight line segments. As the parameters increase, the reconstruction approaches the quadratic case.

If one was to look at a series of reconstructions for increasing values of the parameter T, it would not be difficult to pick out an appropriate value for T, however the goal is to design a heuristic to do this selection automatically. A small value of T reconstructs in straight line segments and is not optimal. As T increases, the reconstructions look better until overshoot starts appearing at the areas of the discontinuities. As T approaches infinity, the reconstruction approaches the reconstruction which using the cubic splines function would create. The most appropriate value of T occurs at the largest value of T, where overshoot, or smoothing, does not occur at the discontinuities.

To implement the heuristic, a method of determining when overshoot occurs is needed. One property of overshoot is that it occurs near areas of large differences between the constraints in the vertical or y-direction. Also, where overshoot occurs, undershoot usually also occurs.

Using these properties, an heuristic can be developed to detect when overshoot occurs. Neighborhoods are first defined to the left and right of large vertical differences between constraints, see Figure 3a. These neighborhoods are simply a group or a window of pixels bordering the large vertical differences. If the sum of the amount of overshoot in one neighborhood, a, and the amount of undershoot in the other neighborhood, b, is greater than 5% of the size of the vertical difference then overshoot is said to occur. That is, overshoot occurs when

$$\frac{a+b}{diff} > 5\%. \tag{12}$$

Because of the similarities between the discontinuity preserving parameter P and the discontinuity preserving parameter T, the above heuristic implementing overshoot detection can be used to determine the optimal values of the parameter P as well.

The parameter λ is the measure of confidence in the data constraints. A small value of λ indicates that the constraints are not good indicators of where the true curve would be, but only indicate the general shape. Alternately, a very large λ indicates that the data constraints accurately predict the location of the true curve. In Figure 2b the curve reconstructed near the data constraints demonstrates the behavior of a large value of λ and the other curve demonstrates what happens when λ is small.

Whereas discontinuity preserving functions are relatively new in the literature and therefore there has not been much done in the area of determining appropriate discontinuity-preserving parameters, the problem of determining the parameter λ introduced in Tikhonov regularization has been previously examined. Cross-validation is a proven method of determining the optimal λ [8, 9]. In general, cross-validation refers to dividing up a data set into two parts and using one subset to assess the performance of a statistical estimator defined on the other. Cross-validation will be of interest here and is described as follows.

For a given λ, let $y^{\bar{c}_i}$ be the curve estimate attained from the constraint points with the i-th constraint point, c_i, omitted. The argument of cross-validation is that if the chosen λ is a good choice for balancing the *a priori* knowledge and the faithfulness to the constraint data in reconstructing the curve, then the i-th location of $y^{\bar{c}_i}$, $y^{\bar{c}_i}(x_i)$, should be a good prediction of the original data point c_i. The fidelity of this prediction is measured as a mean squared error of the difference,

$$\gamma_i^2 = (c_i - y^{\bar{c}_i}(x_i))^2. \tag{13}$$

Figure 3b shows an example of how the error term γ_i, is found.

The error term γ_i, is computed as each of the constraint points are omitted one by one. Cross validation as a function of λ, is defined as:

$$\mathcal{V}(\lambda) = \sum_{i\in\mathcal{C}} \gamma_i^2 = \sum_{i\in\mathcal{C}} (c_i - y^{\bar{c}_i}(x_i))^2 \tag{14}$$

where $\mathcal{C}$ is the collection of constraint point indices. The optimal λ minimizes the functional $\mathcal{V}$.

4.2. Experimental results

In order to test these methods of parameter determination and compare the reconstructions attained using the various signal models, the signal shown in Figure 4a was corrupted by additive Gaussian noise of mean zero and variance one. For each signal model, the appropriate parameters where found using overshoot detection and cross-validation, and the signal was reconstructed. The parameters determined to be optimal in terms of overshoot detection and cross-validation are shown in Table 1 as well as a quantitative measurement of performance in terms of mean absolute error and mean squared error. The corresponding reconstructions are shown in Figures 4b, 4c and 4d. A visual analysis of the reconstructions and the error measurements indicate that the parameters chosen are appropriate.

In the previous paragraph, the optimal parameters were found and the curves reconstructed for a single set of input constraints. For the work presented here, a Monte Carlo simulation is used to evaluate the reconstructions attained from random sets of noisy input constraints, where all sets are generated from the same original signal with the same Gaussian noise characteristics of mean zero, variance one. The signals were reconstructed using the parameters found for a signal input set of constraints, listed in Table 1. Mean absolute error and mean squared error is calculated from each reconsruction. The averages of these errors over the series of reconstructions are of interest here and are also listed in Table 1. Comparing these results with the errors for the reconstructions of a single set of input constraints, an important observation can be made. The errors actually went down slightly between the results listed for a single reconstruction and those listed for the Monte Carlo simulation. This indicates that the optimal parameters selected for one particular set of input constraints can be accurately extended to include any set of input constraints in the same class of sparsity and with the same noise characteristics. This is valuable when the data of a system of interest can be represented in a common expected signal. This signal can then be used to find the optimal parameters for the entire system.

Another interesting test is to explore the behavior of the parameter selection algorithms when the ideas are applied to a real data set. The data set which will be used for this analysis is a section of a line taken from a relatively noise-free image. The image is a black and white image therefore a curve can be defined by plotting pixel location versus pixel intensity. The curve is shown in Figure 5a. This part of the image line was chosen because of the presence of distinct discontinuities. To this data set, zero-mean Gaussian noise with a standard deviation of five is added. This noisy data will be the constraint data used for the reconstructions.

Using the heuristic outlined in the previous section to detect overshoot/undershoot, the discontinuity preserving parameters were found. Similarly, the smoothing parameters for each signal model were found using cross-validation. These optimal parameters are listed in Table 2 as well as the mean absolute error and mean squared error. The corresponding reconstructions are shown in Figures 5b, 5c and 5d.

The real data signal shown in Figure 5 is very different from the original signal used for the previous analysis in that the latter is much smoother. Likewise, the results seen in Table 2 are very different from what was seen before, however, as the reconstructions show, the parameters selected by the algorithms introduced in this work are appropriate.

5. Conclusion

The reconstruction of curves containing discontinuities has many engineering applications. Within this broad problem is the problem of parameter selection. Techniques

Functions	Parameters	MAE	MSE
Cubic Splines	$\lambda = 3$	.007041/ .006783	.000281/ .000264
Generalized Absolute	$\lambda = .1$ $P = 1.2$	.005788/ .005624	.000261/ .000252
Neighborhood Interaction	$\lambda = 1$ $T = 5$	.004893/ .004172	.000106/ .000062
Huber Minimax	$\lambda = .4$ $T = .5$	.003973/ .003687	.000033/ .000026

Table 1: Optimal Parameters and Associated Errors for Synthetic Data Reconstructions for Single Reconstructions/Monte Carlo Reconstructions

Functions	Parameters	MAE	MSE
Cubic Splines	$\lambda = 500$	.024720088	.000750278
Generalized Absolute	$\lambda = 1.0$ $P = 1.3$	.019996250	.000507932
Neighborhood Interaction	$\lambda = 70$ $T = 1.0$	.025116395	.000774358
Huber Minimax	$\lambda = 1.0$ $T = 0.3$	.023467517	.000662282

Table 2: Optimal Parameters and Associated Errors for Real Data Reconstructions

for selecting model parameters for the various methods was proposed. Experiments with the proposed parameter selection techniques indicate that the algorithms accurately determine appropriate parameters for the various methods considered.

References

[1] A. N. Tikhonov and V. Y. Arsenin, *Solutions of Ill-Posed Problems,* V. H. Winston & Sons, Washington, D.C., 1977.

[2] C. H. Reinsch, Smoothing by spline functions, *Numerische Mathematik* **10**, 1967, 177-183.

[3] A. Blake and A. Zisserman, *Visual Reconstruction,* MIT Press, Cambridge, Massachusetts, 1987.

[4] C. Bouman and K. Sauer, An edge-preserving method for image reconstruction from integral projections, in *Proceedings of the Conference on Information Sciences and Systems, Baltimore, Maryland, March 20-22, 1991.*

[5] R. L. Stevenson and E. J. Delp, Fitting curves with discontinuities, in *Proceedings of the IEEE International Workshop on Robust Computer Vision, Seattle, Washington, October 1-3, 1990.*

[6] R. L. Stevenson, B. E. Schmitz and E. J. Delp, Discontinuity Preserving Regularization of Inverse Visual Problems, in *IEEE Transactions on Systems, Man and Cybernetics,* Vol. 24, No. 1, January 1994.

[7] B. E. Schmitz, *Curve Reconstruction: A Balance Between Smoothness and Discontinuity Preservation,* M.S.E.E Thesis, University of Notre Dame, 1993.

[8] G. H. Golub and M. Heath and G. Wahba, Generalized cross-validation as a method for choosing a good ridge parameter, *Technometrics* **21**, 1979, 215-233.

[9] P. Hall and I. Koch, On the feasibility of cross-validation in image analysis, *Society for Industrial and Applied Mathematics - Journal of Applied Mathematics* **52**, 1992, 292-313.

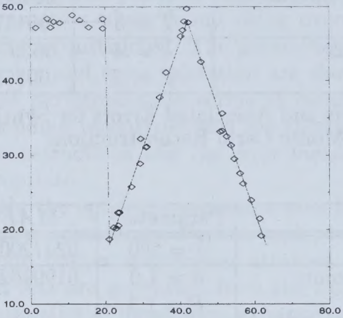
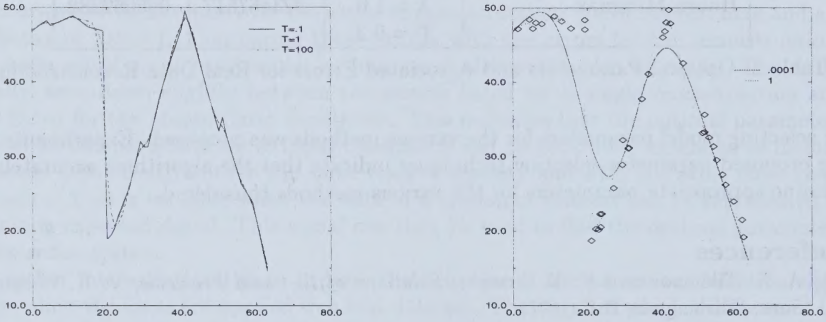


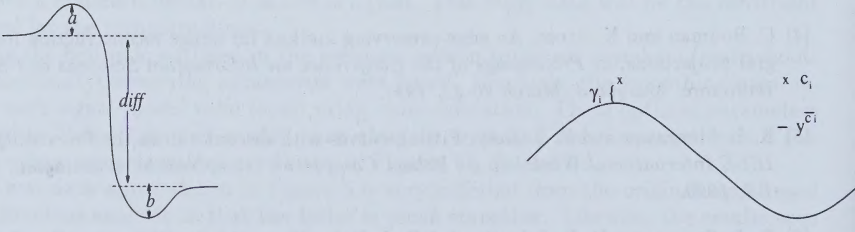
Figure 1: Example Signal for Parameter Investigations



a. Varying T

b. Varying λ

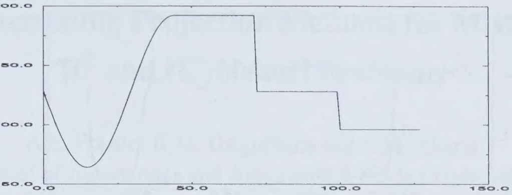
Figure 2: Reconstructions for Parameter Investigations



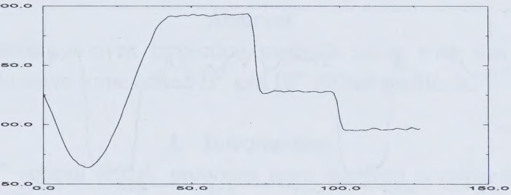
a. Overshoot Detection

b. Cross-Validation

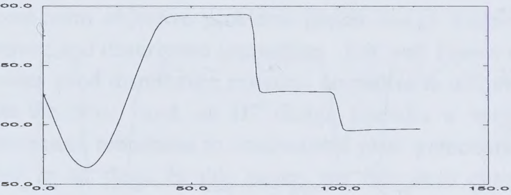
Figure 3: Parameter Selection



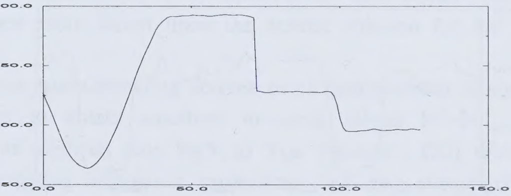
a. Synthetic Data



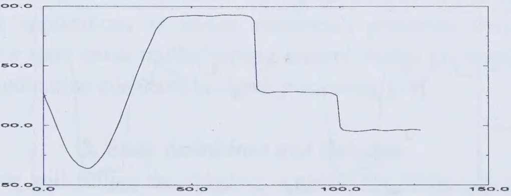
b. Cubic Splines Reconstruction



c. Generalized Absolute Reconstruction

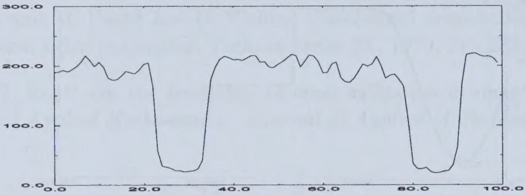


d. Neighborhood Interaction Reconstruction

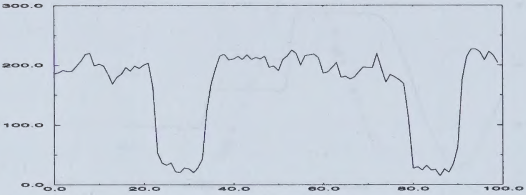


e. Huber Minimax Reconstruction

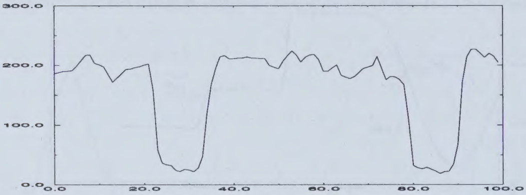
Figure 4: Optimal Reconstructions for Synthetic Data



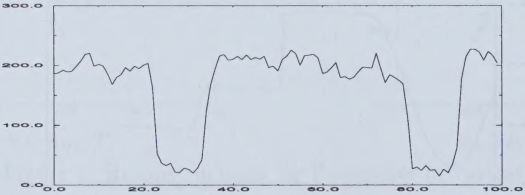
a. Real Data



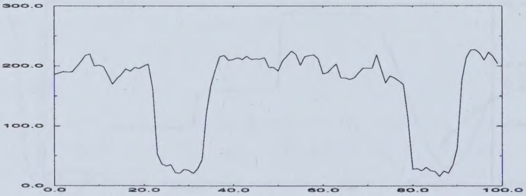
b. Cubic Splines Reconstruction



c. Generalized Absolute Reconstruction



d. Neighborhood Interaction Reconstruction



e. Huber Minimax Reconstruction

Figure 5: Optimal Reconstructions for Real Data

Alternating Projection Methods for Mixed H^2 and H^∞ Nehari Problems

A.E. Frazho, K.M. Grigoriadis and S.M. Kherat
School of Aeronautics and Astronautics, Purdue University
West Lafayette, IN 47907.

Abstract

In this paper, alternating convex projection methods along with fast Fourier transform techniques are used to solve some mixed H^2 and H^∞ Nehari problems.

1. Introduction

Mixed H^2–H^∞ control design problems have received considerable attention lately [3,7,9,12,17,21,22]. The goal of an H^2–H^∞ control problem is to minimize an H^2 performance criterion, subject to a specified H^∞ constraint on the closed loop transfer function. These mixed-norm objective problems permit design trade-offs between root-mean-square performance and disturbance attenuation. It is well known that a Wiener filter (or LQ design), provides good disturbance rejection properties to disturbances with known power spectrum. On the other hand, an H^∞ design provides a worst-case disturbance attenuation which guarantees robustness to unstructured plant perturbations. However, the error in an H^∞ filter is all pass. In this paper, we mix both constraints to yield a compromising result. This work is partially motivated by Kaftal-Larson-Weiss [19] who presented an elegant solution to a suboptimal H^2–H^∞ Nehari problem. For some generalization of their work based upon the central solution for the commutant lifting Theorem see [9],[10].

In this paper, we use alternating convex projection methods along with fast Fourier transform techniques to obtain solutions to some mixed H^2–H^∞ Nehari problems. Alternating projection methods date back to Von Neumann [23] who showed that the sequence of the alternating orthogonal projections onto two subspaces converges to the orthogonal projection onto the intersection of the subspaces. Later, this concept was generalized to the case where the subspaces are replaced by closed convex sets [16]. This result had extensive applications in image restoration problems [24]. Recently, these methods were used for state space multiobjective control design problems [15] and positive Toeplitz matrix approximation problems in signal processing [14].

2. Basic definitions and Notation

In this paper we will follow the standard notation for Hilbert spaces in [1], [8] and [25]. Recall that H^2 is the Hardy space of all analytic functions g in the open unit disc D whose Taylor coefficients are square summable, that is, g is in H^2 if and only if g(z) admits a power series expansion of the form (for z in D)

$$g(z) = \sum_{i=0}^{\infty} g_i z^i \quad \text{and} \quad \|g\|_2^2 = \sum_{i=0}^{\infty} |g_i|^2 < \infty \,. \tag{2.1}$$

Notice that the H^2 norm $\|g\|_2$ of g is defined by the second equation in (2.1). On the other hand, H^∞ is the Hardy space of all uniformly bounded analytic functions in the open unit disc, that is, g is in H^∞ if and only if g is analytic in D and its H^∞ norm defined by,

$$\|g\|_\infty = \sup\{\, |g(z)| \, : z \in D \,\} \tag{2.2}$$

is finite. Obviously, H^∞ is a subspace of H^2.

Let f be any rational function in L^∞. Then the Wiener, or H^2, optimization problem is

$$d_2 = \inf \{\|f - h\|_2 \;:\; h \in H^\infty\} = \inf \|f - H^\infty\|_2 \,. \tag{2.3}$$

Notice that d_2 is the distance from f to H^∞ in the L^2 norm. Moreover, it is easy to show that $d_2 = \|P_- f\|_2$, where P_- is the orthogonal projection onto $K^2 \triangleq L^2 \ominus H^2$. Let g be any function in L^∞. The H^∞ Nehari optimization problem is

$$d_\infty = \inf \{\|g - h\|_\infty \;:\; h \in H^\infty\} = \inf \|g - H^\infty\|_\infty \,. \tag{2.4}$$

Here d_∞ is the distance from g to H^∞ in the L^∞ norm. By the famous Nehari Theorem, $d_\infty = \|\Gamma\|$, where Γ is the Hankel operator from H^2 into K^2 defined by $\Gamma h = P_- gh$ for all h in H^∞. If g is rational, then one can use the state space technique by Glover [13] to compute d_∞, that is, d_∞^2 is the largest eigenvalue of PQ where P is the controllability Gramian and Q is the observability Gramian of a minimal realization for g; see [2,8,12,13] for further details.

We now consider the following mixed H^2–H^∞ problems.

H^2–H^∞ Nehari Optimization Problem (NOP): Given any rational functions f and g in H^∞ and a scalar γ_∞, find an h in H^∞ to solve the following optimization problem.

$$\mu_2 = \inf \{\|f - h\|_2 : h \in H^\infty \text{ and } \|g - h\|_\infty \leq \gamma_\infty \} \,. \tag{2.5}$$

Notice that this optimization problem has a solution if and only if $\gamma_\infty \geq d_\infty$. If $d_\infty = \gamma_\infty$, then the solution to this optimization problem is given by the unique h in H^∞ satisfying $\|g - h\|_\infty = d_\infty$. It is emphasized that d_∞ is not needed to solve the H^2–H^∞ optimization problem (2.5). Our alternating projection techniques will determine whether or not this problem has a solution.

Notice that the H^2–H^∞ optimization problem (2.5) is equivalent to the following problem

$$\min \|f - h\|_2 \qquad \text{subject to} \qquad h \in \mathcal{C}_1 \cap \mathcal{C}_2 \tag{2.6}$$

where $\mathcal{C}_1$ and $\mathcal{C}_2$ are the closed convex sets in L^2 defined by $\mathcal{C}_1 = H^2$ and

$$C_2 = \{\phi \in L^2 \ : \ \|g - \phi\|_\infty \le \gamma_\infty\} . \tag{2.7}$$

Along with the H^2–H^∞ optimization problem in (2.5), let us consider the following feasibility problem:

H^2–H^∞ Nehari Feasibility Problem (NFP): Given the rational functions f and g in L^∞ and the scalars γ_2 and γ_∞, find, if possible, an h in H^∞ satisfying

$$\|f - h\|_2 \le \gamma_2 \qquad \text{and} \qquad \|g - h\|_\infty \le \gamma_\infty \quad (h \in H^\infty) . \tag{2.8}$$

If $\gamma_2 < d_2$, or $\gamma_\infty < d_\infty$, then this feasibility problem does not have a solution. As before, we do not have to compute d_∞ or d_2 to solve this problem. Our alternating projection techniques will determine whether or not a solution exists.

Let C_3 be the closed convex set in L^2 defined by

$$C_3 = \{\phi \in L^2 \ : \ \|f - \phi\|_2 \le \gamma_2\} . \tag{2.9}$$

Then the H^2–H^∞ feasibility problem (2.8) is equivalent to the problem: Find, if possible, a function h such that

$$h \in C_1 \cap C_2 \cap C_3 . \tag{2.10}$$

Throughout, P_i is the orthogonal projection from L^2 onto the convex set C_i for i=1,2,3. Recall that by the projection Theorem, P_i is the (possibly nonlinear) operator defined by $P_i\alpha = \hat{\alpha}$ where α is in L^2 and $\hat{\alpha}$ is the unique element in C_i solving the following optimization problem

$$\| \alpha - \hat{\alpha} \| = \inf \{ \| \alpha - \beta \| \ : \beta \in C_i \} . \tag{2.11}$$

Moreover, $\hat{\alpha} = P_i\alpha$ is the unique element in C_i satisfying the following "maximum principle" condition

$$\operatorname{Re} (\alpha - \hat{\alpha}, \hat{\alpha} - \beta) \ge 0 \qquad (\text{for all } \beta \in C_i) . \tag{2.12}$$

Our procedure in solving the optimization problem (2.6) and the feasibility problem (2.8) is to use alternating projection methods on the convex sets described by C_1, C_2 and C_3. For a function α in L^2, let $\{\alpha_n\}_{n=-\infty}^{\infty}$ be the coefficients of the Fourier series expansion of α, that is

$$\alpha(z) = \sum_{n=-\infty}^{\infty} \alpha_n z^n \qquad (\text{for } z = e^{it}) . \tag{2.13}$$

The orthogonal projection $P_1 = P_+$ onto $C_1 = H^2$ is the linear operator on L^2 defined by

$$P_1\alpha = \sum_{n=0}^{\infty} \alpha_n z^n \qquad (\alpha \in L^2) . \tag{2.14}$$

It is easy to show (see for instance example 1.4.1 in [1]) that the orthogonal projection P_2 onto C_2 can be computed as follows.

Proposition 2.1. *Let* g *be in* L^2 *and let* C_2 *be the ball of radius* γ_∞ *with center* g *defined by (2.7). Then the orthogonal projection* P_2 *onto* C_2 *is the nonlinear operator defined by*

$$(P_2\alpha)(e^{it}) = \begin{cases} \dfrac{\alpha(e^{it}) - g(e^{it})}{|\alpha(e^{it}) - g(e^{it})|}\gamma_\infty + g(e^{it}) & \text{if } |\alpha(e^{it}) - g(e^{it})| > \gamma_\infty \\ \alpha(e^{it}) & \text{otherwise} \end{cases} \tag{2.15}$$

where α *is in* L^2.

The orthogonal projection P_3 onto C_3 is the nonlinear operator given by

$$P_3\alpha = \begin{cases} \dfrac{\alpha - f}{\|\alpha - f\|_2}\gamma_2 + f & \text{if } \|f - \alpha\|_2 > \gamma_2 \\ \alpha & \text{otherwise} \end{cases} \tag{2.16}$$

where α is in L^2.

3. The Alternating Convex Projections Method

In this section we will review the alternating convex projection techniques in [4,5,16,18]. Then, we will use these alternating projection techniques along with fast Fourier transform techniques to compute a solution for the H^2–H^∞ Nehari optimization problem in (2.5) and the H^2–H^∞ Nehari feasibility problem in (2.8). To begin, let $\{C_1, C_2, ..., C_m\}$ be a family of closed convex sets in a Hilbert space $\mathcal{H}$, and C be the closed convex set defined by $C = C_1 \cap C_2 \cap \cdots \cap C_m$. Let P_i be the orthogonal projection onto the set C_i and P_C the orthogonal projection operator onto C. Now define the composition operator Φ on $\mathcal{H}$ by $\Phi \triangleq P_n P_{n-1} ... P_1$. It follows that Φ is a contraction and the convex set C is precisely the set of fixed points of Φ, see [5]. The following will be used to solve our feasibility problem.

Theorem 3.1 ([5]). *Let* x_o *be any vector in* $\mathcal{H}$. *If* C *is not empty, then* $\Phi^i x_o$ *converges weakly to a vector* x *in* C *as* $i \to \infty$. *If* C *is empty, the sequence* $\{x_i\}_{i=0}^\infty$ *of alternating projections* $x_1 = P_1 x_0$, $x_2 = P_2 x_1$, ..., $x_n = P_n x_{n-1}$, $x_{n+1} = P_1 x_n$, ... *does not converge.*

Notice that if C is empty, then $\Phi^i x_0$ may converge. However, the sequence $\{x_i\}_{i=0}^\infty$ will not converge. For example, if one has two disjoint balls in the plane, then the sequence $\{x_i\}_{i=0}^\infty$ will oscillate between two limit points, and $\Phi^i x_0$ will converge. If C is not empty and the sets C_i are affine manifolds, then the sequence $\{\Phi^i x_o\}_{i=0}^\infty$ converges weakly to the orthogonal projection $P_C x_o$ of x_o onto C, that is, $x = P_C x_o$. On the other hand, if the C_i's are general convex sets and C is not empty, then the limit x does not necessarily equal $P_C x_o$. For example, consider the simple case where C_1 is a disc in $\mathbb{R}^2$ and C_2 is a line through the center of the disk. However, a simple modification of the previous alternating projection technique can be used to compute the orthogonal projection $P_C x_o$; see [4,18]. This result is summarized in the following theorem which is used to solve our optimal H^2–H^∞ Nehari problem.

Theorem 3.2 ([4]). *Assume that C is not empty. Let x_o be any vector in $\mathcal{H}$ and define the sequence $\{x_i\}_{i=0}^{\infty}$ by*

$$\begin{array}{l}
\left.\begin{array}{lcl}
x_1 = P_1 x_o & , & z_1 = x_1 - x_o \\
x_2 = P_2 x_1 & , & z_2 = x_2 - x_1 \\
\vdots & & \vdots \\
x_n = P_n x_{n-1} & , & z_n = x_n - x_{n-1}
\end{array}\right\} 1^{st}\text{ cycle} \\
\left.\begin{array}{lcl}
x_{n+1} = P_1(x_n - z_1) & , & z_{n+1} = x_{n+1} - (x_n - z_1) \\
x_{n+2} = P_2(x_{n+1} - z_2) & , & z_{n+2} = x_{n+2} - (x_{n+1} - z_2) \\
\vdots & & \vdots \\
x_{2n} = P_n(x_{2n-1} - z_n) & , & z_{2n} = x_{2n} - (x_{2n-1} - z_n)
\end{array}\right\} 2^{nd}\text{ cycle} \\
\begin{array}{lcl}
x_{2n+1} = P_1(x_{2n} - z_{n+1}) & & z_{2n+1} = x_{2n+1} - (x_{2n} - z_{n+1}) \\
\vdots & & \vdots
\end{array}
\end{array} \tag{3.1}$$

Then the sequence $\{x_i\}_{i=0}^{\infty}$ converges weakly to the orthogonal projection $P_C x_o$ of x_o onto C. The convex set C is empty if and only if the sequence $\{x_i\}_{i=0}^{\infty}$ does not converge.

The algorithm provided by Theorem 3.2 can be used to solve our H^2–H^∞ Nehari optimization problem in (2.5). These algorithms can be implemented very easily in a computer, but may suffer from slow convergence (for example, consider the case where C_1 and C_2 are hyperplanes with "small" angles between them). One way to overcome this problem is to make use of some "directional information" in the algorithm. For simplicity of presentation we will state this algorithm for two convex sets, that is $C = C_1 \cap C_2$. This result is used to give us a fast algorithm to solve our feasibility problem. By iterating on γ_2, this algorithm can also be used to provide a solution for our H^2–H^∞ Nehari optimization problem.

Theorem 3.3 ([16]). *Assume that $C = C_1 \cap C_2$ is not empty. Let x_o be any vector in $\mathcal{H}$ and define the sequence $\{x_i\}_{i=0}^{\infty}$ by*

$$\left.\begin{aligned} &x_1 = P_1 x_o\,,\ \ x_2 = P_2 x_1\,,\ \ x_3 = P_1 x_2 \\ &x_4 = x_1 + \lambda_1 (x_3 - x_1)\,,\ \ \lambda_1 = \frac{\| x_1 - x_2 \|^2}{(\,x_1 - x_3,\, x_1 - x_2\,)} \end{aligned}\right\}\ 1^{\text{st}}\text{ cycle}$$

$$\left.\begin{aligned} &x_5 = P_1 x_4\,,\ \ x_6 = P_2 x_5\,,\ \ x_7 = P_1 x_6 \\ &x_8 = x_5 + \lambda_2 (x_7 - x_5)\,,\ \ \lambda_2 = \frac{\| x_5 - x_6 \|^2}{(\,x_5 - x_7,\, x_5 - x_6\,)} \end{aligned}\right\}\ 2^{\text{nd}}\text{ cycle} \tag{3.2}$$

$$\vdots$$

Then the sequence $\{x_i\}_{i=0}^{\infty}$ *converges weakly to a vector* x *in* C. *The set* C *is empty if and only if the sequence* $\{x_i\}_{i=0}^{\infty}$ *does not converge.*

Notice that each cycle of the previous algorithm utilizes the projections onto the sets C_1 and C_2 to obtain a point which is closer to the intersection $C_1 \cap C_2$ than the point obtained from a cycle of Theorem 3.1.

Remark 3.4. The result of Theorem 3.3 can be carried over to the case of three sets C_1, C_2 and C_3 as follows. Given a point x_0 in $\mathcal{H}$, compute the first cycle of the directional algorithm in Theorem 3.2 for two sets, say C_1 and C_2, to obtain a point x_1. Then, starting from x_1, compute the directional projection cycle for the sets C_2 and C_3 to obtain a point x_2. Follow the same procedure for the sets C_3 and C_1 and repeat the directional projection cycle for the pairs C_1, C_2, then C_2, C_3, and finally C_3, C_1, alternatively, until the sequence $\{x_i\}_{i=0}^{\infty}$ converges.

4. The Mixed H^2–H^∞ Alternating Projections Algorithm

In this section, we present some explicit formulae involving the fast Fourier transform to compute the orthogonal projections P_1, P_2 and P_3. These expressions, along with the previous alternating projection algorithms, will allow us to compute a solution to our H^2–H^∞ optimization problems.

Numerically, for any rational function α in L^2, the orthogonal projections $P_1\alpha$, $P_2\alpha$ and $P_3\alpha$ can be computed, approximately, to any accuracy, using fast Fourier transform techniques. To this end, suppose that $\alpha = n/d$, where $n(z) = n_0 + n_1 z + \ldots + n_{m-1} z^{m-1}$ and $d(z) = d_0 + d_1 z + \ldots + d_{m-1} z^{m-1}$ are complex polynomials of $z = e^{it}$. Choose, a number $K = 2^j$ (where j is integer and $K >> 2m$) of evaluation points around the unit circle, and compute the following vectors, using FFT s:

$$[\hat{n}_0, \hat{n}_1, \ldots, \hat{n}_{K-1}] = \mathrm{FFT}([n_0, n_1, \ldots, n_{m-1}], K) \tag{4.1}$$

$$[\hat{d}_0, \hat{d}_1, \ldots, \hat{d}_{K-1}] = \mathrm{FFT}([d_0, d_1, \ldots, d_{m-1}], K) \tag{4.2}$$

where $\hat{n}_k = n(e^{i2\pi k/K})$, $\hat{d}_k = d(e^{i2\pi k/K})$, $k = 0, 1, \ldots, K-1$ and $\hat{x} = \mathrm{FFT}(x, K)$ denotes the fast Fourier transform of length K of a vector x padded with the appropriate number of zeros. Divide n/d, pointwise around the unit circle, to obtain

$$\alpha(e^{i2\pi k/K}) = \hat{\alpha}_k = \hat{n}_k/\hat{d}_k, \qquad (k = 0, 1, ..., K-1)\,. \tag{4.3}$$

The Fourier coefficients of $\alpha = n/d$ can be approximated by computing the inverse FFT

$$[\alpha_0, \alpha_1, ..., \alpha_{K-1}] = \mathrm{IFFT}([\hat{\alpha}_0, \hat{\alpha}_1, ..., \hat{\alpha}_{K-1}], K) \tag{4.4}$$

where IFFT(y,K) denotes the inverse fast Fourier transform of length K of the vector y. Hence, we have

$$\alpha \approx \alpha_0 + \alpha_1 z + \alpha_2 z^2 + ... + \frac{\alpha_{K-1}}{z} + \frac{\alpha_{K-2}}{z^2} + ... \tag{4.5}$$

where the summation converges to α in L^2 norm, when $K \to \infty$. Notice that the first half of the coefficients $\alpha_0, ..., \alpha_{K/2}$ correspond to the H^2 part of α while the last half of the coefficients $\alpha_{K-1}, \alpha_{K-2}, ..., \alpha_{K/2+1}$ correspond to the $L^2 \ominus H^2$ part of α.

Computation of the projection $\beta \triangleq P_1\alpha$: The Fourier coefficients of the orthogonal projection $\beta = P_1\alpha$ are obtained by setting to zero the last $K/2 - 1$ elements of the vector $[\alpha_0, \alpha_1, ..., \alpha_{K-1}]$ in (4.4), i.e.,

$$\beta \triangleq P_1\alpha \approx \sum_{k=0}^{K/2} \alpha_k z^k \qquad (z = e^{it}) \tag{4.6}$$

where the summation in (4.6) approaches $\beta = P_1\alpha$ in the L^2 norm when $K \to \infty$. Hence,

$$[\hat{\beta}_0, \hat{\beta}_1, ..., \hat{\beta}_{K-1}] = \mathrm{FFT}([\alpha_0, \alpha_1, ..., \alpha_{K/2}, 0, ..., 0], K) \tag{4.7}$$

where $\hat{\beta}_k = \beta(e^{i2\pi k/K})$.

Computation of the projection $\beta = P_2\alpha$: Let $\alpha = n/d$ and $g = p/q$ be given rational functions in L^2, where $p(z) = p_0 + p_1 z + ... + p_{m-1} z^{m-1}$ and $q(z) = q_0 + q_1 z + ... + q_{m-1} z^{m-1}$. The orthogonal projection $\beta = P_2\alpha$, provided by Proposition 2.1, can be obtained as follows: First compute the vector $\hat{g} = [\hat{g}_0, \hat{g}_1, ..., \hat{g}_{K-1}]$, where $g(e^{i2\pi k/K}) = \hat{g}_k$. This is done by computing $\hat{g}_k = \hat{p}_k/\hat{q}_k$, where

$$[\hat{p}_0, \hat{p}_1, ..., \hat{p}_{K-1}] = \mathrm{FFT}([p_0, p_1, ..., p_{m-1}], K) \tag{4.8}$$

$$[\hat{q}_0, \hat{q}_1, ..., \hat{q}_{K-1}] = \mathrm{FFT}([q_0, q_1, ..., q_{m-1}], K)\,. \tag{4.9}$$

The projection $\beta = P_2\alpha$ can be computed pointwise on the unit circle as follows. Let $r = \gamma_\infty$ and define

$$\hat{\eta}_k = \left[\frac{1}{2}\left(|\hat{\alpha}_k - \hat{g}_k| - r - \big|\,|\hat{\alpha}_k - \hat{g}_k| - r\,\big|\right) + r\right](\hat{\alpha}_k - \hat{g}_k) \quad \text{for } k = 0, ..., K-1,$$

then

$$\hat{\beta}_k = \frac{\hat{\eta}_k}{|\hat{\alpha}_k - \hat{g}_k| + \varepsilon} + \hat{g}_k \quad \text{for } k = 0, ..., K-1 \tag{4.10}$$

where ε is an arbitrary small positive number to avoid numerical problems when the difference $\hat{\alpha}_k - \hat{g}_k$ is close to zero. The Fourier coefficients β_i of the orthogonal projection $\beta = P_2\alpha$ are obtained by $[\beta_0, \beta_1, ..., \beta_{K-1}] = \mathrm{IFFT}([\hat{\beta}_0, \hat{\beta}_1, ..., \hat{\beta}_{K-1}])$, i.e,

$$\beta = P_2\alpha \approx \beta_0 + \beta_1 z + \beta_2 z^2 + ... + \frac{\beta_{K-1}}{z} + \frac{\beta_{K-2}}{z^2} + ... \qquad (z = e^{it}) \tag{4.11}$$

where the summation approaches $P_2\alpha$ in the L^2 norm when $K \to \infty$. Finally, we used equation (4.10) rather than the equivalent formula (2.15) because equation (4.10) allows to bypass a loop statement in our computer program which saves some computational time.

Computation of the projection $\beta = P_3\alpha$: Let $f = r/s$. The computation of the orthogonal projection $P_3\alpha$ of α onto the set C_3, provided by (2.16), requires to compute the distance $\|f - \alpha\|_2$. For this, we first compute the vector $[\hat{f}_0, \hat{f}_1, ..., \hat{f}_{K-1}]$ which contain the values of the function f around the unit circle. This vector can be obtained by fast Fourier transform techniques, as described earlier. Next, using the inverse FFT we compute the vector $[\phi_0, \phi_1, ... , \phi_{K-1}] = \mathrm{IFFT}([\hat{\alpha}_0 - \hat{f}_0, \hat{\alpha}_1 - \hat{f}_1, ..., \hat{\alpha}_{K-1} - \hat{f}_{K-1}], K)$. By Parseval's formula, the distance $\|f - \alpha\|_2$ is provided by the Euclidean norm of the vector $[\phi_0, \phi_1, ..., \phi_{K-1}]$, that is,

$$\|f - \alpha\|_2^2 \approx \sum_{k=0}^{K-1} |\phi_k|^2 . \tag{4.12}$$

Now, the Fourier coefficients β_i of the projection $\beta = P_3\alpha$ can be computed using equation (2.16), as follows

$$\beta_k = \begin{cases} \dfrac{\alpha_k - f_k}{\|\alpha - f\|_2}\gamma_2 + f_k & \text{if } \|f - \alpha\|_2 > \gamma_2 \\ \alpha_k & \text{otherwise} \end{cases} \tag{4.13}$$

for k = 0 , ... ,K − 1. Hence,

$$P_3\alpha \approx \beta_0 + \beta_1 z + \beta_2 z^2 + ... + \frac{\beta_{K-1}}{z} + \frac{\beta_{K-2}}{z^2} + ... \qquad (z = e^{it}) \tag{4.14}$$

where the summation in (4.14) approaches $P_3\alpha$ in the L^2 norm. The evaluation points $\hat{\beta}_k = \beta(e^{i2\pi k/K})$ are provided by

$$[\hat{\beta}_0, \hat{\beta}_1, ..., \hat{\beta}_{K-1}] = \mathrm{FFT}([\beta_0, \beta_1, ..., \beta_{K-1}], K) . \tag{4.15}$$

The solution to the H^2–H^∞ Nehari optimization problem (2.5) is obtained by applying the projection algorithm of Theorem 3.2. To this end, we consider the sets C_1 and C_2, and the initial point $x_0 = f$. Then the sequence of projections of the Algorithm 3.1 converges to the solution h of the H^2–H^∞ Nehari optimization problem.

The solution to the H^2–H^∞ Nehari feasibility problem (2.8) can be computed using the alternating projection algorithm of Theorem 3.1 or the directional projection algorithm of Theorem 3.3. The required constraint sets $C_1 = H^2$ and C_2 and C_3 are defined by (2.7) and (2.9). Starting from any initial point x_0, the above algorithms are guaranteed to converge to a feasible solution h, if and only if the intersection $C_1 \cap C_2 \cap C_3$ is nonempty. Moreover, by iterating on γ_2, one can determine the smallest γ_2 such that the intersection $C_1 \cap C_2 \cap C_3$ is not empty, and a corresponding h in this intersection. This h is the unique solution to the optimization problem in (2.5) and the smallest γ_2 is μ_2; see (2.5). Now let $h = [h_0, h_1, h_2, ...] = \mathrm{IFFT}(\hat{h}_0, \hat{h}_1, ...)$ be the solution to our H^2–H^∞ optimization problem obtained by the alternating projection technique. Then apply the Kalman-Ho algorithm [6],[20] to the Markov parameters $\{h_i\}$ to obtain a minimal realization $\{A, B, C\}$ for h, that is, $h_i = CA^iB$ for $i \geq 0$. This realization provides a rational approximation for our solution h for the H^2–H^∞ Nehari problem.

Finally, it is noted that given the rational functions $f_1, f_2, ..., f_n$ and $g_1, g_2, ..., g_n$ in L^∞, and the scalars $\gamma_{21}, ..., \gamma_{2n}, \gamma_{\infty 1}, ..., \gamma_{\infty n}$, our technique can be used to find, if it exists, an h in H^∞ that satisfies

$$\|f_j - h\|_2 \leq \gamma_{2j} \qquad \text{and} \qquad \|g_k - h\|_\infty \leq \gamma_{\infty k}$$

where j=1,2,...,n, and k=1,2,...,n.

REFERENCES

[1] A. V. Balakrishnan, *Applied Functional Analysis*, Second Edition, Springer-Verlag, New York, 1981.

[2] J. A. Ball, I. Gohberg and L. Rodman, *Interpolation for Rational Matrix Functions*, Birkhauser-Verlag, Basel, 1990.

[3] D.S. Bernstein and W.M. Haddad, "LQG Control with an H_∞ Performance Bound: A Riccati Equation Approach," *IEEE Trans. Autom. Contr.*, Vol. 34, No.3, pp. 293-305, 1989.

[4] J. P. Boyle, and R. L. Dykstra, "A Method for Finding Projections onto the Intersection of Convex Sets in Hilbert Space," Advances in Order Restricted Statistical Inference, Vol. 37, Lecture Notes in Statistics, Springer-Verlag, pp. 28-47, 1986.

[5] W. Cheney, and A. A. Goldstein, "Proximity maps for convex sets," Proc. Amer. Math. Soc., Vol. 12, pp. 448-450, 1959.

[6] A.A.H Damen, P.M.J Vanden Hof and A.K. Hajdasinski, "Approximate realization based upon an alternative to the Hankel matrix: the page matrix", *Systems and Control Letters,* **2** (1982), pp.202-208.

[7] J.C. Doyle, B.A. Francis and A.R. Tannenbaum, *Feedback Control Theory,* MacMillan Publishing Co., New York, 1992.

[8] C. Foias and A. E. Frazho, *The Commutant Lifting Approach to Interpolation Problems,* Operator Theory Advances and Applications, **44**, Birkhauser-Verlag, Basel, 1990.

[9] C. Foias and A. E. Frazho, Commutant Lifting and Simultaneous H^∞ and L^2 Suboptimization, *SIAM J. Math. Anal.*, **23**, No. 4, (1992), pp. 984-994.

[10] C. Foias, A. E. Frazho, and W. S. Li, The Exact H^2 Estimate for the Central H^∞ Interpolant, *Integral Equations and Operator Theory,* to appear.

[11] B.A. Francis, *A Course in H_∞ Control Theory,* Springer-Verlag, Lecture notes in Control and Information Sciences, Vol.88, New York, 1987.

[12] A.E. Frazho and S.M. Kherat,"Applications of the Schur Algorithm to Mixed H^2–H^∞ Nehari Problems," *J. Mathematical Analysis and Applications,* to appear.

[13] K. Glover, "All Optimal Hankel-Norm Approximations of Linear Multivariable Systems and their L^∞-Error bounds," *Int. J. Control,* Vol.39, No.6, pp. 1115-1193, 1984.

[14] K. M. Grigoriadis, A.E. Frazho and R. E. Skelton, "Application of Alternating Convex Projection Methods for Computation of Positive Toeplitz Matrices," *IEEE Trans. Signal Proc.,* to appear.

[15] K. M. Grigoriadis and R. E. Skelton, "Alternating Convex Projection Methods for Covariance Control Design," Proc. 30th Allerton Conf. Commun. Control Computing, Monticello, Illinois, 1992. Also, *Int. J. Control,* to appear.

[16] L. G. Gubin, B. T. Polyak and E. V. Raik, "The Method of Projections for Finding the Common Point of Convex Sets," *USSR Computational Mathematics and Mathematical Physics,* Vol. 7, pp. 1-24, 1967.

[17] W.M. Haddad, D.S. Bernstein and D. Mustafa, "Mixed H_2/H_∞ Regulation and Estimation: The Discrete-Time Case," *Systems and Control letters,* Vol.16, pp. 235-247, 1991.

[18] S-P. Han, "A Successive Projection Method", *Mathematical Programming,* Vol. 40, pp. 1-14, 1988.

[19] V. Kaftal, D. Larson, and G. Weiss, Quasitriangular subalgebras of semifinite Von Neumann algebras are closed, *J. Functional Analysis,* **107**, No. 2 (1992), pp. 387-401.

[20] R.E. Kalman, P.L. Falb and M.A. Arbib, *Topics in Mathematical System Theory,* McGraw Hill, New York, 1969.

[21] I. Kaminer, P.P Khargonekar and M.A. Rotea, "Mixed H_2/H_∞ Control for Discrete-Time Systems via Convex Optimization," *Automatica,* **29**, No.1 (1993), pp.57-70.

[22] P. P. Khargonekar and M. A. Rotea, Mixed H^2/H^∞ Control: A Convex Optimization Approach, *IEEE trans. Autom. Control,* **36**, No 7, (1991), pp. 824-837.

[23] J. Von Neumann, *Functional Operators,* vol II, Ann. Math. Studies No.22, Princeton, NJ, 1950.

[24] D. C. Youla, and H. Webb, "Image restoration by the method of convex projections: Part 1 - Theory," *IEEE Trans. Medical Imag.,* Vol. 1, No. 2, pp. 81-94, 1982.

[25] N. Young, *An Introduction to Hilbert Space,* Cambridge University Press, Cambridge, NY, 1988.

$\mathcal{H}_2$ and $\mathcal{H}_\infty$ Optimal Realizations of Finite Wordlength Digital Filters[*]

Mario A. Rotea
School of Aeronautics and Astronautics
Purdue University
West Lafayette, IN 47907-1282.

Darrell Williamson
Department of Engineering
Australian National University
Canberra, ACT Australia.

Abstract

The problem of finding realizations that minimize the sensitivity to quantization noise of a finite wordlength implementation of a discrete-time linear state-space filter subject to scaling of the filter internal signals is considered. Finite wordlength implementations which invoke quantization either before or after multiplication, and possibly include integer residue feedback, are represented in a unified framework. This framework is used to pose and solve four different optimization problems that yield realizations that minimize the output roundoff noise gain subject to overflow or scaling constraints. Optimal realizations are derived based on either an $\mathcal{H}_2$ or $\mathcal{H}_\infty$ roundoff noise gain subject to either $\mathcal{H}_2$ or $\mathcal{H}_\infty$ scaling constraints on the input-to-internal-signal transfer matrices. One of the optimization problems has a well known analytical solution, the other three are reduced to the problem of minimizing a linear function subject to affine matrix inequality constraints, which is a convex optimization problem whose global optimum may be readily found. This solution, together with the unified framework for the analysis of several FWL implementations, constitutes the main contribution of this paper.

1 Introduction

The synthesis of digital filters in fixed point format is important in high speed signal processing applications. In particular, when the bandwidth of the filters are narrow, recursive (or IIR) filters offer computational advantages over nonrecursive implementations. However, IIR filters can suffer severely from the effects of finite wordlength (FWL) arithmetic [9, 5, 1, 13, 6] unless one selects an appropriate structure. In other applications such as Kalman filtering and digital control, a recursive filter is mandatory, see [14].

Let $F(z)$ denote the $p \times m$ transfer matrix of the given filter, and consider the set of minimal state-space realizations of $F(z)$ defined by

$$\begin{aligned} \mathcal{R}(F) := \{(A,B,C,D) &\in R^{n\times n} \times R^{n\times m} \times R^{p\times n} \times R^{p\times m} : \\ F(z) &= C(zI_n - A)^{-1}B + D \text{ and } (A,B,C) \text{ is minimal } \}. \end{aligned}$$

(Here, R denotes the real numbers.) Clearly, given any $(A,B,C,D) \in \mathcal{R}(F)$, the discrete-time system, which we denote by $\mathcal{F}$, may be implemented by solving the state-space equation

$$\mathcal{F} := \left\{ \begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) + Du(k)\,, \end{aligned} \right. \tag{1.1}$$

where $u(k) \in R^m$ denotes the input of the filter, $y(k) \in R^p$ is the output, and $x(k) \in R^n$ is the filter state.

In the ideal case of infinite arithmetic precision, any two realizations from the set $\mathcal{R}(F)$ used in (1.1) will produce the same sequences of filtered outputs in response to any given input sequence and zero initial conditions. This is not the case when (1.1) is implemented with FWL

---

[*] Supported in part by the National Science Foundation under RIA Grant no. ECS-9108493, and in part by the Cooperative Research Centre for Robust and Adaptive Systems at ANU.

arithmetic. This is simply because both the coefficients of the matrices (A, B, C, D) and the internal signals (e.g., state components) in (1.1) must be quantized to finite wordlength.

Coefficient quantization corresponds to rounding each coefficient to the nearest value based on the available wordlength. On the other hand, *roundoff signal quantization* consists of rounding every internal signal at each time instant k to the nearest available quantization level. If any internal signal exceeds the available dynamic range, then *arithmetic overflow* occurs. The usual practice is then to employ saturation arithmetic whereby the particular offending variable is set to the maximum allowable magnitude. Overflow can cause significant distortion; thus, the filter internal signals must be scaled so as to restrict the possibility of overflow occurrence.

These FWL effects cause a degradation in the performance of the filter which depends on the particular realization used in (1.1). In this paper, we study the problem of finding realizations in $\mathcal{R}(F)$ that minimize the performance degradation at the filter output due to fixed point roundoff quantization of the internal signals only, subject to scaling or overflow constraints.

The paper has four more sections. In Section 2, we give a general representation of several FWL implementations most commonly used in practice. This representation is our first contribution and it permits all subsequent results to be developed in a unified framework.

In Section 3, we define the optimization problems that will be used to select suitable realizations. Four optimization problems are introduced and they are referred to as the *optimal FWL synthesis problems.* The function to be minimized in the FWL synthesis problems is the so-called roundoff noise gain and it is measured in terms of either an $\mathcal{H}_2$ or an $\mathcal{H}_\infty$ norm. The $\mathcal{H}_2$ noise gain measures the departure from the ideal filter response (i.e. no signal quantization) when the quantization residual has fixed and known power spectral density (PSD); the $\mathcal{H}_\infty$ noise gain measures *worst-case* deviation from the ideal response when the PSD of the quantization residual is unknown. The scaling constraints are given in terms of either the $\mathcal{H}_2$ or the $\mathcal{H}_\infty$ norm on the transfer matrices from the filter input to the each one of the filter internal signals. An $\mathcal{H}_2$ scaling constraint limits the size of an internal signal when the PSD of the filter input is precisely known. On the other hand, an $\mathcal{H}_\infty$ scaling constraint limits the *maximum* possible size of an internal signal over all inputs with bounded power.

In Theorem 3.1 we show how to reduce the FWL synthesis problems to four *auxiliary optimization problems*, which are easier to solve than the original problems. In the case of $\mathcal{H}_2$ scaling constraints, a solution to the auxiliary problem yields a solution to the original FWL synthesis problem. In the case of $\mathcal{H}_\infty$ scaling constraints, the auxiliary problem is equivalent to the minimization of an upper bound for the roundoff noise gain subject to multiple $\mathcal{H}_\infty$ scaling constraints.

The auxiliary (and the original) optimization problem with $\mathcal{H}_2$ roundoff noise gain and $\mathcal{H}_2$ scaling has a well-known closed form solution [9, 5, 13, 6]. In Section 4, we give the solution to the other three auxiliary problems. In Theorem 4.1 we show how all these problems may be reduced to the *convex optimization problem of minimizing a linear function subject to affine matrix inequality constraints.* We believe this is an important contribution of our work because recent developments in convex optimization show that global solutions to these problems are obtained with a very reasonable computational effort; see, for example, [3, 12]. Finally, the conclusions of our work are summarized in Section 5. This conference paper is an abridged version of our recent paper [10], where the interested reader may find more details and proofs of the results.

We now briefly introduce notation which is used in this paper. Given a real matrix A, trace A denotes its trace, $\det A$ denotes its determinant, and A' its transpose. I_n denotes the $n \times n$ identity matrix; without subindex, I denotes an identity matrix whose dimension depends on the context. Given real matrices $A = A'$ and $B = B'$, we say $A < B$ (respectively, $A \leq B$) iff the difference $B - A$ is positive-definite (respectively, positive semidefinite). We make use of the forward shift operator $\sigma(\cdot)$ defined by $(\sigma x)(k) := x(k+1)$. The standard $\mathcal{H}_2$ and $\mathcal{H}_\infty$ norms are denoted by $\|\cdot\|_2$ and $\|\cdot\|_\infty$, respectively; these norms are defined in the usual way.

2 Unified Representation for FWL Filters

Most fixed point FWL implementations of the state-space equation (1.1) can be described by the feedback interconnection shown in Figure 1. In this figure, $\hat{\mathcal{F}}$ is a linear shift invariant system while $\mathcal{N}$ is a memoryless nonlinear system that represents the quantizer. This interconnection is described by the equations

$$\begin{aligned} \hat{\mathcal{F}} := &\begin{cases} \sigma\hat{x} &= A\hat{x} + B_0(A)e + Bu \\ \hat{z} &= C_0(A)\hat{x} + D_0(B)u \\ \hat{y} &= C\hat{x} + D_1(C)e + Du \end{cases} \\ \mathcal{N} := &\{\; e = Q[\hat{z}] - \hat{z}, \end{aligned} \tag{2.1}$$

where $\hat{x}$ and $\hat{y}$ are the state and the output respectively, of the fixed point FWL implementation, while e represents the quantization error and $\hat{z}$ is the internal signal being quantized. Here, $Q[\cdot]$ is a nonlinear operator that implements roundoff quantization, of each component of the vector $\hat{z}(k)$, to t fractional bits.

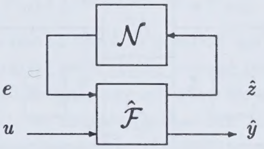


Figure 1: Representation of FWL implementations.

The matrix functions $B_0(A)$ and $C_0(A)$ are *affine* functions of A, while the functions $D_0(B)$ and $D_1(C)$ are *linear* in their arguments. These functions depend on the specific fixed point FWL implementation used and they satisfy the following property: Given any nonsingular matrix $T \in R^{n\times n}$, we have

$$\begin{aligned} &B_0(T^{-1}AT) = T^{-1}B_0(A)T\ ; && C_0(T^{-1}AT) = T^{-1}C_0(A)T \\ &D_0(T^{-1}B) = T^{-1}D_0(B)\ ; && D_1(CT) = D_1(C)T. \end{aligned} \tag{2.2}$$

The above property shall be assumed throughout this paper.

To illustrate how the representation in Figure 1 arises, let us first suppose that the FWL implementation of the state equation (1.1) employs arithmetic quantization *before multiplication*. In this case, the implementation is described by the nonlinear equation

$$\sigma\hat{x} = AQ[\hat{x}] + Bu, \quad \hat{y} = CQ[\hat{x}] + Du. \tag{2.3}$$

Define now the quantization error

$$e := Q[\hat{x}] - \hat{x}. \tag{2.4}$$

Simple manipulations show that (2.3) and (2.4) are described by (2.1) with $B_0(A) := A$, $C_0(A) := I_n$, $D_0(B) := 0$, $D_1(C) := C$.

In some cases, the performance of the implementation (2.3) may be improved by using *integer residue feedback* [13], which is also referred to as δ-operator implementation [8]. In this case, the fixed point FWL implementation is described by

$$\sigma\hat{x} = AQ[\hat{x}] + Bu - e, \quad \hat{y} = CQ[\hat{x}] + Du, \tag{2.5}$$

where e is defined by (2.4); equivalently, (2.5) may be represented by

$$\sigma\hat{x} = \hat{x} + (A - I_n)Q[\hat{x}] + Bu, \quad \hat{y} = CQ[\hat{x}] + Du.$$

Again, we conclude that (2.5) may be described by (2.1) with $B_0(A) := A - I_n$, $C_0(A) := I_n$, $D_0(B) := 0$, $D_1(C) := C$.

In [10] we showed that the feedback interconnection of Figure 1 describes many others fixed point FWL implementations of practical interest. In the sequel, we will use this representation to analyze the performance of all these FWL implementations in a unified framework.

3 Optimal FWL Filter Synthesis Problems

We first make use of the general representation given in (2.1) (see also Figure 1) to give a representation of the *error system*, i.e. the system generating the difference between the ideal filter output y and the output $\hat{y}$ of the given FWL implementation.

Let $(A, B, C, D) \in \mathcal{R}(F)$ denote an initial state-space realization. Any other realization in $\mathcal{R}(F)$ is then of the form $(T^{-1}AT, T^{-1}B, CT, D)$ for some nonsingular similarity transformation T. Suppose we use this latter realization in the ideal state-space model (1.1) and its FWL implementation (2.1). Let u denote a given input sequence applied to both (1.1) and (2.1), and define the output error signal

$$z := \hat{y} - y\,, \tag{3.6}$$

where $\hat{y}$ is output of the FWL implementation. Then using some simple algebra and the identities in (2.2), it follows from (1.1) and (2.1) that z is generated by the feedback interconnection shown in Figure 2.

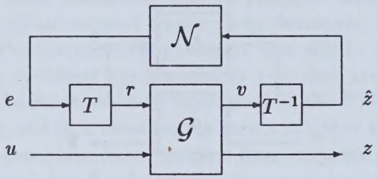


Figure 2: Representation of error system.

In this figure, the quantization error e is as defined in (2.1), i.e. $e = Q[\hat{z}] - \hat{z}$, T is the similarity transformation introduced earlier, and $\mathcal{G}$ is a shift invariant error system with state-space equation:

$$\mathcal{G} = \left\{ \begin{array}{rcl} \sigma\psi & = & A\psi + B_0(A)r \\ \sigma\xi & = & A\xi + B_0(A)r + Bu \\ v & = & C_0(A)\xi + D_0(B)u \\ z & = & C\psi + D_1(C)r\,, \end{array} \right. \tag{3.7}$$

where $\psi = T(\hat{x} - x)$ and $\xi = T\hat{x}$. In the sequel, we let G denote the transfer matrix of the error system $\mathcal{G}$. From (3.7) it readily follows that G has the triangular structure

$$G = \begin{bmatrix} G_{vr} & G_{vu} \\ G_{zr} & 0 \end{bmatrix}, \tag{3.8}$$

where G_{vr} is the transfer matrix from r to v, G_{vu} is the transfer matrix from u to v and G_{zr} is the transfer matrix from r to z. Note that G is a function of the initial state-space realization only, and does not depend on T.

The dependence of the quantized signal $\hat{z}$, and the output error signal z, on the similarity transformation T is complex due to the presence of the nonlinear block $\mathcal{N}$, see Figure 2. In signal processing, it is customary to remove $\mathcal{N}$ and assume that the quantization error e is an *exogenous signal.* This is the so-called *signal + noise* model of the quantization process [7]. There is some analytical justification [11] for this assumption, and simulation studies have verified its usefulness.

Using this signal + noise model, the performance degradation due to the quantization noise e of the given realization may be measured using the so-called *roundoff noise gain* of the error system. In this paper we consider a roundoff gain $\phi_p(T)$ as defined by

$$\phi_p(T) := \|G_{zr}TW_e\|_p^2. \tag{3.9}$$

(Note that the transfer matrix from u to z is zero.) In (3.9), W_e denotes a stable transfer matrix with stable inverse representing a *frequency dependent quantization error weight.* This weighting function is incorporated to add flexibility into the problem formulation. The subindex "p" in (3.9) denotes the *performance norm* of interest. In this paper, the norm is either the $\mathcal{H}_2$ norm ($p = 2$) or the $\mathcal{H}_\infty$ norm ($p = \infty$).

Minimization of $\phi_p(T)$ over the set of nonsingular similarity transformations T is trivial. In fact, from (3.9) we get that that $\phi_p(T)$ can be made arbitrarily small by making T arbitrarily small. However, making T arbitrarily small will lead to numerical overflow problems. This is simply because the transfer matrix from the input of the filter u to the quantized signal $\hat{z}$ is $T^{-1}G_{vu}$. Hence, if T is arbitrarily small, $\hat{z}$ is an arbitrarily large signal, and the FWL implementation will overflow even for small inputs u.

As a consequence, one should only consider the minimization of $\phi_p(T)$ subject to an *overflow or scaling constraint.* An overflow constraint is imposed on the transfer matrix from u to the i-th component of $\hat{z}$. We consider the following overflow measures:

$$\psi_{i,q}(T) := \|e_i'T^{-1}G_{vu}W_u\|_q^2. \tag{3.10}$$

(Here, e_i is the unit vector with a one in the i-th position and zeros elsewhere.) The weighting function W_u is a stable transfer matrix with stable inverse representing a frequency dependent input weight, and the subindex "q" denotes the *overflow norm* of interest. As in the definition of the roundoff noise gain in (3.9), we consider either the $\mathcal{H}_2$ norm ($q = 2$) or the $\mathcal{H}_\infty$ norm ($q = \infty$). The aim is then to minimize the roundoff noise gain $\phi_p(T)$ in (3.9) subject to constraints on the overflow measures $\psi_{i,q}(T)$ in (3.10). We shall use two different formulations depending on the choice of the overflow norm.

This optimization problem, which we call the *(p, q) optimal FWL synthesis problem*, is precisely defined as follows: "Let p, q, and a positive number γ, be given. Compute the optimal constrained roundoff gain $\nu_{p,q}$ defined by

$$\nu_{p,q} := \begin{cases} \inf\{\phi_p(T) : T \text{ s.t. } \det T \neq 0 \text{ and } \psi_{1,2}(T) = \psi_{2,2}(T) = \ldots = \psi_{n,2}(T) = \gamma\} & \text{if } q = 2 \\ \inf\{\phi_p(T) : T \text{ s.t. } \det T \neq 0 \text{ and } \psi_{1,\infty}(T) \leq \gamma, \ldots, \psi_{n,\infty}(T) \leq \gamma\} & \text{if } q = \infty \end{cases} \tag{3.11}$$

and, given any $\epsilon > 0$, find a similarity transformation T (which depends on p and q) that satisfies the appropriate constraints in (3.11), and $\phi_p(T) < \nu_{p,q} + \epsilon$."

3.1 Signal Processing Interpretations

The optimal FWL synthesis problem defined above has several interesting interpretations depending on the choice of norms p and q.

Case (i): $p = 2$ and $q = 2$. Suppose that the filter input u is a (wide sense) stationary stochastic signal with known autocorrelation matrix $R_u(k) = E\{u(k)u'(0)\}$ and power spectral density (PSD)

$$S_u(\theta) = \sum_k R_u(k)e^{-j\theta k}.$$

Then, in signal processing, it is customary to assume (provided u is sufficiently exciting [11]) that the quantization error sequence e defined in (2.1) is a zero mean stationary white noise process uncorrelated with u and with PSD

$$S_e(\theta) = \sigma_e^2 I, \ \ \sigma_e^2 := \frac{1}{12}2^{-2t},$$

where t is the fractional wordlength of each component of $\hat{z}_i$ [7]. Applying the signals e and u to the error system in Figure 2 (without the block $\mathcal{N}$), we then conclude that the outputs $\hat{z}$ and z are also stationary stochastic processes with power spectral densities given by

$$S_{\hat{z}}(\theta) = G_{\hat{z}e}(e^{j\theta})S_e(\theta)G_{\hat{z}e}(e^{-j\theta}) + G_{\hat{z}u}(e^{j\theta})S_u(\theta)G_{\hat{z}u}(e^{-j\theta}) \tag{3.12}$$

and

$$S_z(\theta) = G_{ze}(e^{j\theta})S_e(\theta)G_{ze}(e^{-j\theta}). \tag{3.13}$$

where we have defined $G_{\hat{z}e} := T^{-1}G_{vr}T$, $G_{\hat{z}u} := T^{-1}G_{vu}$, and $G_{ze} := G_{zr}T$, to avoid carrying the dependence on T in the subsequent development.

Now, if a spectral factor W_u is introduced for S_u (e.g. $S_u = W_uW_u^*$), and we define $W_e = \sigma_e I_n$, it follows from the spectral densities in (3.12) and (3.13) that the variance of each component of the quantized signal $\hat{z}$ is given by $E\{\hat{z}_i^2\} = \|e_i'G_{\hat{z}e}W_e\|_2^2 + \|e_i'G_{\hat{z}u}W_u\|_2^2$. It is common practice to neglect the first term in this equation since the weight W_e is very small when compared with the input spectral factor W_u. Thus, to a good approximation, we obtain

$$E\{\hat{z}_i^2\} \approx \|e_i'G_{\hat{z}u}W_u\|_2^2 =: \psi_{i,2}(T). \tag{3.14}$$

Similarly, from (3.13), the variance of the output error z is

$$E\{z'z\} = \|G_{ze}W_e\|_2^2 =: \phi_2(T). \tag{3.15}$$

Hence, the minimization of the variance of the output error signal z, subject to the scaling constraints on the variances of the quantized signals $\hat{z}_1 \ldots \hat{z}_n$ (e.g. $\psi_{i,2}(T) = \gamma$), is the FWL synthesis problem defined in (3.11) with $p = q = 2$. This is a classical problem which was solved in [9, 5, 13].

Case (ii): $p = 2$ and $q = \infty$. Suppose now that: (i) the PSD of the quantization error e is fixed and known, say $S_e = W_eW_e^*$, (ii) u and e are uncorrelated, and (iii) the PSD of the input signal u satisfies the condition:

$$\frac{1}{2\pi}\,\mathrm{trace}\int_0^{2\pi} W_u^{-1}(e^{j\theta})S_u(\theta)W_u^{-1}(e^{-j\theta})\,d\theta \leq 1, \tag{3.16}$$

where W_u is a known function representing the prior knowledge on S_u. That is, u is no longer fixed and known, but is generated by passing an arbitrary stationary stochastic process (with variance bounded by 1) through the known filter W_u.

Then, under the assumption that e and u are uncorrelated, using well-known properties of the $\mathcal{H}_\infty$ norm, we have from (3.12) and (3.16) that $E\{\hat{z}_i^2\} \leq \|e_i'G_{\hat{z}e}W_e\|_2^2 + \|e_i'G_{\hat{z}u}W_u\|_\infty^2$. Moreover, the right hand side is the smallest upper bound for $E\{\hat{z}_i^2\}$ that holds for arbitrary stochastic processes u satisfying (3.16). Once again, if the first term in this last expression is small, we obtain the approximation

$$E\{\hat{z}_i^2\} \leq \|e_i'G_{\hat{z}u}W_u\|_\infty^2 =: \psi_{i,\infty}(T). \tag{3.17}$$

Note also that, since $S_e = W_eW_e^*$ is fixed and known, the variance of z is still given by (3.15).

Thus, if all we know about the input u is that it satisfies (3.16), from (3.15) and (3.17), we conclude that to minimize the performance degradation, subject to an overflow constraint, one should solve the FWL synthesis problem with $p = 2$ and $q = \infty$.

In [10], we gave further details on the interpretations of the FWL synthesis problems. In addition, [10] discusses the importance of using the $\mathcal{H}_\infty$ roundoff noise gain and/or $\mathcal{H}_\infty$ scaling when there is uncertainty on the spectral properties of the quantization error e and/or the input u.

3.2 Auxiliary Optimization Problems

In order to solve the FWL synthesis problems we make use of four auxiliary optimization problems, which we now introduce. First, define the *single* overflow measure:

$$\psi_q(T) := \|T^{-1}G_{vu}W_u\|_q^2. \tag{3.18}$$

The (p,q) *auxiliary FWL synthesis problem* is defined as follows: "Let p and q be given. Compute the optimal constrained roundoff gain $\mu_{p,q}$ defined by

$$\mu_{p,q} := \inf\{\phi_p(T) : T \text{ such that } \det T \neq 0 \text{ and } \psi_q(T) \leq 1\} \tag{3.19}$$

and, given any $\epsilon > 0$, find a nonsingular similarity transformation T (depending on p and q) that satisfies $\psi_q(T) \leq 1$ and $\phi_p(T) < \mu_{p,q} + \epsilon$. "

In the sequel, we shall refer to a solution T as introduced in the definitions of optimization problems (3.11) and (3.19) as an "ϵ-optimal solution" to the appropriate optimization problem. This notation streamlines the presentation of our results, and it will also be used for any other optimization problem we introduce. The next result shows the connection between the original problem (3.11) and the auxiliary problem (3.19) and it is the main result of this section.

Theorem 3.1 *The FWL synthesis problem (3.11) and the auxiliary FWL synthesis problem (3.19) satisfy the following:*

1. **Case** $q = 2$: *Suppose the weight* $W_e = w_e I_n$, *where* $w_e(z)$ *is a scalar function, then*

$$\nu_{p,2} = \frac{1}{n\gamma}\,\mu_{p,2}.$$

Furthermore, if T denotes an $(n\gamma\epsilon)$-optimal solution to the auxiliary problem (3.19), there exists a real orthogonal matrix U such that

$$\hat{T} := \sqrt{\frac{\psi_2(T)}{n\gamma}}\,TU$$

satisfies $\psi_{1,2}(\hat{T}) = \ldots = \psi_{n,2}(\hat{T}) = \gamma$ *and* $\phi_p(\hat{T}) < \nu_{p,2} + \epsilon$. *That is,* $\hat{T}$ *is an* ϵ*-optimal solution for the FWL synthesis problem (3.11).*

2. **Case** $q = \infty$: *Let T denote a $(\gamma\epsilon)$-optimal solution to the auxiliary problem (3.19), then*

$$\hat{T} := \sqrt{\frac{\psi_2(T)}{\gamma}}\,T$$

satisfies $\psi_{1,\infty}(\hat{T}) \leq \gamma \ldots \psi_{n,\infty}(\hat{T}) \leq \gamma$ *and* $\phi_p(\hat{T}) < \frac{1}{\gamma}\,\mu_{p,\infty} + \epsilon$. *Hence,* $\nu_{p,\infty} \leq \frac{1}{\gamma}\,\mu_{p,\infty}$.

This result shows that when $\mathcal{H}_2$ scaling is used (i.e. $q = 2$) and the weight W_e is a scalar transfer function, a solution to the FWL synthesis problem (3.11) may be obtained from a solution to the auxiliary problem (3.19) for both $\mathcal{H}_2$ ($p = 2$) and $\mathcal{H}_\infty$ ($p = \infty$) roundoff noise gains. When $\mathcal{H}_\infty$ scaling constraints are considered (i.e. $q = \infty$), the auxiliary problem only gives an upper bound for the original optimal roundoff gain $\nu_{p,\infty}$. In this sense, the auxiliary problem (3.19) is a conservative approach to the FWL synthesis problem (3.11) when $q = \infty$.

A proof of Theorem 3.1 is in [10]. The proof in [10] also indicates how the orthogonal matrix U, required to construct a solution to the FWL synthesis problem (3.11) with $q = 2$, may be computed. This construction is standard [4, 5], and is based on the fact that given any similarity transformation T, there exists a real orthogonal matrix U such that the overflow measures defined in (3.10) and (3.18) satisfy $\psi_{1,2}(TU/\sqrt{n}) = \psi_{2,2}(TU/\sqrt{n}) = \ldots = \psi_{n,2}(TU/\sqrt{n}) = \psi_2(T)$.

4 A solution via convex programming

In the rest of this paper we make the following assumption.

Assumption 4.1 *The weighting function W_e introduced in (3.9) is of the form $W_e = w_e I_n$, where $w_e(z)$ is a stable scalar transfer function.*

To give our main result, we need to introduce a few definitions. Let G be given by (3.8) and define the transfer matrices $H_1 := G_{zr} W_e$ and $H_2 := G_{vu} W_u$. Take stabilizable and detectable state-space realizations for H_1 and H_2, say

$$H_j(z) = C_j(zI - A_j)^{-1} B_j + D_j, \; j = 1, 2. \tag{4.20}$$

One can find such realizations from the state-space representation of the error system $\mathcal{G}$ in (3.7) and state-space formulae for the weights. Note that A_1 and A_2 have all eigenvalues in $|z| < 1$.

Let L_o and L_c denote the observability gramian of the pair (C_1, A_1) and the controllability gramian of the pair (A_2, B_2), respectively; i.e. L_o and L_c are the unique solution of the Lyapunov equations

$$A_1' L_o A_1 + C_1' C_1 = L_o \text{ and } A_2 L_c A_2' + B_2 B_2' = L_c.$$

Define the matrices S_1 and S_2 by

$$S_1 := B_1' L_o B_1 + D_1' D_1 \text{ and } S_2 := C_2 L_c C_2' + D_2 D_2'. \tag{4.21}$$

Given a real number ρ, and symmetric matrices $X = X'$, $Y = Y'$ and $Z = Z'$, define the following matrix-valued functions:

$$\mathcal{C}_1(X, Z) := \begin{bmatrix} A_1 & B_1 \\ C_1 & D_1 \end{bmatrix}' \begin{bmatrix} X & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} A_1 & B_1 \\ C_1 & D_1 \end{bmatrix} - \begin{bmatrix} X & 0 \\ 0 & Z \end{bmatrix} \tag{4.22}$$

$$\mathcal{C}_2(Y, Z) := \begin{bmatrix} A_2 & B_2 \\ C_2 & D_2 \end{bmatrix} \begin{bmatrix} Y & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} A_2 & B_2 \\ C_2 & D_2 \end{bmatrix}' - \begin{bmatrix} Y & 0 \\ 0 & Z \end{bmatrix} \tag{4.23}$$

$$\mathcal{C}_3(\rho, X, Z) := \begin{bmatrix} A_1 & B_1 \\ C_1 & D_1 \end{bmatrix} \begin{bmatrix} X & 0 \\ 0 & Z \end{bmatrix} \begin{bmatrix} A_1 & B_1 \\ C_1 & D_1 \end{bmatrix}' - \begin{bmatrix} X & 0 \\ 0 & \rho I \end{bmatrix} \tag{4.24}$$

Note that these are affine functions of their arguments. The next result gives a state-space solution to the auxiliary FWL problem.

Theorem 4.1 *Consider the auxiliary FWL problem defined in (3.19) and suppose that Assumption 4.1 holds. The optimal performance $\mu_{p,q}$ of the (p, q) auxiliary FWL optimal synthesis problem is given by the value of the optimization problems in Table 1. Moreover, an ϵ-optimal solution $Z_{p,q} = Z_{p,q}'$ of the optimization problem in entry (p, q) of Table 1 is positive definite, and any nonsingular real matrix $T_{p,q}$ that satisfies*

$$T_{p,q} T_{p,q}' = \begin{cases} Z_{2,2}^{-1} & \text{if } p = 2 \text{ and } q = 2 \\ (\mu_{\infty,2} + \epsilon)\, Z_{\infty,2}^{-1} & \text{if } p = \infty \text{ and } q = 2 \\ Z_{p,\infty} & \text{if } q = \infty \end{cases}$$

is an ϵ-optimal solution to the (p, q) auxiliary FWL synthesis problem in (3.19).

A proof of Theorem 4.1 is in [10]. The optimization problems in Table 1 are *convex* problems involving affine matrix inequalities. Thus, there are effective interior point algorithms [3, 12], for computing the auxiliary optimal performance $\mu_{p,q}$, and finding ϵ-optimal solutions. When $p = q = 2$, a numerical search is not required for a closed form solution is available. This closed form solution was derived, for example, in [9, 5, 13].

In [2] a convex optimization problem similar to the one in entry $(p = 2, q = \infty)$ of Table 1 has been proposed for solving a *special case* of the $(2, \infty)$ auxiliary FWL problem. This special case corresponds to taking

$$H_1(z) = C(zI_n - A)^{-1} \text{ and } H_2(z) = (zI_n - A)^{-1} B$$

Norm	$q = 2$	$q = \infty$
$p = 2$	$\mu_{2,2} = \inf_Z \ \mathrm{trace} S_1 Z^{-1}$ s.t. $Z = Z' > 0$ $\mathrm{trace} S_2 Z \leq 1$	$\mu_{2,\infty} = \inf_{(Y,Z)} \ \mathrm{trace} S_1 Z$ s.t. $Y = Y' > 0$ $Z = Z'$ $C_2(Y, Z) < 0$
$p = \infty$	$\mu_{\infty,2} = \inf_{(X,Z)} \ \mathrm{trace} S_2 Z$ s.t. $X = X' > 0$ $Z = Z'$ $C_1(X, Z) < 0$	$\mu_{\infty,\infty} = \inf_{(\rho,X,Y,Z)} \ \rho$ s.t. $X = X' > 0$ $Y = Y' > 0$ $Z = Z'$ $C_2(Y, Z) < 0$ $C_3(\rho, X, Z) < 0$

Table 1: Solution to the auxiliary FWL synthesis problem.

in (4.20), where (A, B, C) denotes the initial state-space realization of the filter. Equivalently, this corresponds to taking $B_0(A) = I_n$, $C_0(A) = I_n$, $D_0(B) = 0$, and $D_1(C) = 0$ in our general representation (2.1), and weights $W_e = I_n$ and $W_u = I_m$. To solve this problem, [2] proposes a convex program where the optimization variables are three symmetric matrices of size n, this means a total of $\frac{3}{2}\, n(n + 1)$ optimization variables. As it can be seen from the table above, our convex program involves a search for two $n \times n$ symmetric matrices; thus, we only have $n(n+1)$ optimization variables. Moreover, our formulation has two affine matrix inequalities, the first one ($Y > 0$) of size n and the second one ($C_2(Y, Z) < 0$) of size $2n$ since, in this special case, $C_2 = I_n$ in (4.23). In [2] one finds three affine matrix inequalities, two of the same dimension than ours; the third one, of dimension $n + m$ where m is the number of inputs to the filter. Hence, we also have less number of constraints. Since the program in [2] and the one in Table 1 both have a linear objective function, we conclude that our formulation is simpler.

5 Conclusions

The problem of finding state-space realizations that minimize the output roundoff noise gain of a FWL implementation of a discrete time linear filter subject to scaling of the filter internal signals has been considered. Solutions were obtained for several FWL implementations of practical interest. This was made possible by the introduction of a filter representation that describes a number of FWL implementations in a unified way.

Four different optimization problems to quantify the performance, as measured by a roundoff noise gain, versus overflow tradeoff have been introduced. Both $\mathcal{H}_2$ and $\mathcal{H}_\infty$ roundoff noise gains, and $\mathcal{H}_2$ and $\mathcal{H}_\infty$ scaling or overflow constraints were considered, giving rise to the $\mathcal{H}_p/\mathcal{H}_q$ optimal FWL synthesis problem. The $\mathcal{H}_2$ roundoff gain should be used when the spectral characteristics of the quantization error sequence e are fixed and known. Otherwise, one should use the $\mathcal{H}_\infty$ roundoff gain as this yields a realization that is robust to perturbations in the power spectral density of e; i.e. the performance of the filter is guaranteed provided that e has bounded variance or power. Similarly, the $\mathcal{H}_2$ scaling should be used when the spectral characteristics of the filter input u are fixed and known. If this assumption is not satisfied, then one should use $\mathcal{H}_\infty$ scaling as this limits the maximum possible size of the internal signals when u is only known to have bounded power.

To solve the FWL synthesis problems we have introduced four auxiliary problems. In the case of $\mathcal{H}_2$ scaling, a solution to the auxiliary problem yields a solution to the original optimization problem. When the scaling is given by $\mathcal{H}_\infty$ constraints, the auxiliary problems yield upper bounds on the original optimal performance. The $\mathcal{H}_2/\mathcal{H}_2$ auxiliary problem has a known analytical solution. We have shown that the other auxiliary problems reduce to the problem of minimizing a linear function subject to affine matrix inequality constraints, which is a *solvable* convex optimization problem. This solution, together with the unified framework for the analysis of several FWL implementations, constitutes the main contribution of this paper.

References

[1] R.C. Agarwal and C.S. Burrus, “New recursive digital filter structures having very low sensitivity and roundoff noise”, *IEEE Trans. Circuits Syst.*, Vol. CAS-22, no 12, pp 921-927, Dec. 1975.

[2] S. Boyd, V. Balakrishnan, E. Feron, and L. El Ghaoui, “Control system analysis and synthesis via linear matrix inequalities,” *Proc. of the 1993 American Control Conference, San Francisco*, CA, Vol. 2, pp. 2147- 2154, 1993.

[3] S. Boyd and L. El Ghaoui, “Method of centers for minimizing generalized eigenvalues,” preprint, 1992.

[4] G. Golub and C.F. Van Loan. *Matrix Computations.* The Johns Hopkins University Press, 1989.

[5] S.Y. Hwang, “Minimum uncorrelated unit noise in state space digital filtering”, *IEEE Trans. on Acoustics, Speech, and Signal Processing*, Vol. ASSP-25, pp 273-281, Aug. 1977.

[6] G. Li and M. Gevers, “Roundoff noise minimization using delta operator realizations”, *IEEE Trans. on Acoustics, Speech and Signal Processing*, Feb. 1993.

[7] B. Liu, “Effect of finite wordlength on the accuracy of digital filters - a review,” *IEEE Trans.on Circuit Theory*, Vol. CT-18, pp 670-677, Nov. 1976.

[8] R.H. Middleton and G.C. Goodwin, *Digital Control and Estimation*, Prentice Hall, 1990.

[9] C.T. Mullis and R.A. Roberts, “Synthesis of minimum roundoff noise fixed-point digital filters”, *IEEE Trans. Circuits Syst.*, Vol. CAS-23, pp 551-562, Sept. 1976.

[10] M. A. Rotea and D. Williamson, "Optimal synthesis of FWL filters via affine matrix inequalities," submitted to *IEEE Trans. Circuits Syst.*, June 1993.

[11] A. Sripad and D.L. Snyder, “A necessary and sufficient condition for quantization errors to be uniform and white”, *IEEE Trans. on Acoustics, Speech, and Signal Processing*, Vol. ASSP-25, pp 442-448, Oct. 1977.

[12] L. Vandenberghe and S. Boyd, “Primal-dual potential reduction method for problems involving matrix inequalities,” preprint, 1993.

[13] D. Williamson, “Roundoff noise minimization and pole-zero sensitivity in fixed point digital filters using residue feedback”, *IEEE Trans. Acoustics, Speech and Signal Processing*, Vol. ASSP-34, no 5, pp. 1210-1220, Oct. 1986.

[14] D. Williamson, *Digital Control and Implementation*, Prentice Hall, 1991.

Eliminating Pole/Zero Cancellations in MIMO $\mathcal{H}^\infty/\mathcal{H}^2$ Control using Interpolation *

K. C. Goh and M. G. Safonov
Department of Electrical Engineering - Systems,
University of Southern California, Los Angeles, CA 90089-2563, U.S.A.

Abstract

Two-Riccati $\mathcal{H}^\infty/\mathcal{H}^2$ controller synthesis tends to produce controllers which cancel plant open left half plane poles and zeros in the closed loop, even in the sub-optimal $\mathcal{H}^\infty$ case. This often makes the $\mathcal{H}^\infty/\mathcal{H}^2$ designs unusable when the plant to be controlled has left half plane poles or zeros near the $j\omega$-axis.

In this paper, we use matrix Nevalinna-Pick tangential interpolation theory to generate auxiliary compensators which will modify the 2-Riccati controller so as to selectively eliminate pole-zero cancellations near the $j\omega$-axis. The order of the resulting controller will not be greater than the order of the original 2-Riccati controller and the resulting design will tend to be considerably more robust.

1 Introduction

Given a plant $G(s)$, and a prescribed $\gamma > 0$, the two-Riccati solution to the regular $\mathcal{H}^\infty$ problem [7, 4, 20] parameterizes all stabilizing controllers $K_Q(s)$ such that $\| \mathcal{LFT}[G(s), K_Q(s)] \|_\infty < \gamma$ where $K_Q(s)$ is given in the form of a lower linear fractional transformation of a central controller, $K(s)$, and any $m_2 \times r_2$ auxiliary compensator, $Q(s)$, i.e. $K_Q(s) := \mathcal{LFT}[K(s), Q(s)]$, subject to $Q(s)$ being asymptotically stable and having an infinity norm less than γ.

A particularly disturbing characteristic of the 2-Riccati $\mathcal{H}^\infty$ central controller is the well known and often observed tendency of the controller to have left half plane poles (respectively zeros) which "cancel" the left half plane zeros (respectively poles) of the plant in the closed loop. This is very undesirable near the $j\omega$-axis because slight plant perturbations will often then result in an unstable closed loop. There are numerous examples of this occuring in the $\mathcal{H}^\infty$ applications literature, [2] for example, particularly in the mixed sensitivity design framework [19] and for flexible structures.

References [11, 18, 17] contain model matching / matrix fraction description analyses of the phenomenon. The two-Riccati solution to the $\mathcal{H}^\infty$ problem manifests the same pole-zero cancellation problem, even for the sub-optimal case, and the structural reasons for this have been analyzed in [21] and most recently in [9] which showed that the pole-zero cancellations will remain regardless of the value of γ or the choice of $Q(s)$.

The remedies which have appeared in the literature [18, 22, 23] are all basically based on the not totally satisfactory concept of changing the problem (i.e. the plant or the

---

*Research supported in part by AFOSR grant F49620-92-J-0014

problem formulation) to suit the solution (i.e. 2-Riccati $\mathcal{H}^\infty$ synthesis). In this paper, we offer an alternative approach which focuses on using the addtional degree of freedom afforded by the auxiliary compensator $Q(s)$ to modify the 2-Riccati $\mathcal{H}^\infty$ controller, while preserving the closed loop stability and infinity norm. We note that LMI-based techniques, e.g. [5], also offer an alternative approach to $\mathcal{H}^\infty$ synthesis.

In particular, the problem of generating a $Q(s)$ so as to eliminate particular controller/plant pole/zero cancellations near the $j\omega$-axis is equivalent to one of finding a $Q(s)$ so as to cause internal pole-zero cancellations *within* $K_Q(s)$ at those particular points. This is shown to be convertible into a matrix Nevalinna-Pick tangential interpolation problem for which a closed form analytic solution is available. Our results enable us to synthesize analytically a solution to the $\mathcal{H}^\infty$ problem which does not give closed loop pole-zero cancellations near the $j\omega$-axis, and which has order no greater than the order of the plant.

Although we will mainly concentrate on $\mathcal{H}^\infty$ synthesis, our results are also applicable to sub-optimal $\mathcal{H}^2$ design as well as the singular $\mathcal{H}^\infty/\mathcal{H}^2$ problem.

2 Preliminaries

We will consider the finite dimensional linear time-invariant plant

$$G(s) = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix} \overset{ss}{=} \left[\begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{array}\right] \tag{1}$$

where $A \in \mathcal{R}^{n\times n}$, $D_{11} \in \mathcal{R}^{m_1 \times r_1}$, $D_{22} \in \mathcal{R}^{m_2 \times r_2}$ and $G_{ij}(s) := D_{ij} + C_i(sI - A)^{-1}B_j$.

We will assume the following conditions:

Assumption 1 *$\{A,B_2,C_2\}$ is stabilizable and detectable.*

Assumption 2 *$r_2 \le m_1$ and $m_2 \le r_1$.*

Assumption 3 $\begin{bmatrix} -sI + A & B_2 \\ C_1 & D_{12} \end{bmatrix}$ *and* $\begin{bmatrix} -sI + A^T & C_2^T \\ B_1^T & D_{21}^T \end{bmatrix}$ *have no zeros on the $j\omega$-axis and both D_{12} and D_{21}^T have full column rank.*

Let $K_Q(s)$ denote the controller to be synthesized, with descriptor form representation $K_Q(s) \overset{des}{=} \left[\begin{array}{c|c} -sE_{K_Q} + A_{K_Q} & B_{K_Q} \\ \hline C_{K_Q} & D_{K_Q} \end{array}\right]$. Define the combined system

$$T_{y_1u_1}(s) := \mathcal{LFT}[G(s), K_Q(s)] \overset{des}{=} \left[\begin{array}{c|c} -sE_T + A_T & B_T \\ \hline C_T & D_T \end{array}\right] :=$$

$$\left[\begin{array}{cc|c} -sI + A + B_2\check{D}D_{K_Q}C_2 & B_2\check{D}C_{K_Q} & B_1 + B_2\check{D}D_{K_Q}D_{21} \\ B_{K_Q}\hat{D}C_2 & -sE_{K_Q} + A_{K_Q} + B_{K_Q}\hat{D}D_{22}C_{K_Q} & B_{K_Q}\hat{D}D_{21} \\ \hline C_1 + D_{12}D_{K_Q}\hat{D}C_2 & D_{12}\check{D}C_{K_Q} & D_{11} + D_{12}\check{D}D_{K_Q}D_{21} \end{array}\right] \tag{2}$$

where $\check{D} := (I - D_{K_Q}D_{22})^{-1}$ and $\hat{D} := (I - D_{22}D_{K_Q})^{-1}$.

Definition 1 (Regular Sub-Optimal $\mathcal{H}^\infty$ Control Problem) *For some $\gamma \in \mathcal{R}_+$, determine a controller $K_Q(s)$ for which $\| T_{y_1u_1}(s) \|_\infty < \gamma$ and $-sE_T + A_T$ has no zeros in $\bar{\mathcal{C}}^+$.*

To simplify the formulae involved, we impose the additional assumption:

Assumption 4 $D_{11} = 0$.

The results for the case $0 \leq \bar{\sigma}(D_{11}) < \gamma$ are given in in [9].

The following notation is used: " $\overset{des}{=}$ " denotes "descriptor form", " $\overset{ss}{=}$ " denotes "state space form" and " $\mathcal{LFT}[G(s), K_Q(s)]$ " denotes "the lower linear fractional transformation of the $G(s)$ by $K_Q(s)$ as defined by (2)". Let $\mathcal{C}^{oe}$, $\mathcal{C}^+$, $\bar{\mathcal{C}}^+$, $\mathcal{C}^-$ denote the extended $j\omega$-axis, the open, closed right half, and open left half complex planes respectively. For brevity, with a slight abuse of terminology, the phrase "uncontrollable/unobservable modes of $G(s)$" shall refer to the uncontrollable/unobservable modes of the state space representation of a plant (A, B, C, D) which has transfer function $G(s) := D + C(sI - A)^{-1}B$.

Matrix pencils, [6, 24, 12, 13], are order one polynomial matrices of the form $P(s) = -sP_E + P_A$, where P_E and P_A are real matrices. If $P(s_o)\xi_o = 0$ for some finite $s_o \in \mathcal{C}$, and $P(s)\xi_o \neq 0$ for all finite complex $s \neq s_o$, then we term s_o a *zero* of $P(s)$ and say that $P(s)$ has a *eigenvector* ξ_o.

3 The Structure of the 2-Riccati $\mathcal{H}^\infty$ Solution

The 2-Riccati solution to the $\mathcal{H}^\infty$ problem may be formulated in matrix pencil/descriptor form, [24, 12], (the LQG case) and [3]. Define the Hamiltonian Pencils for the plant (1):

$$W_{12}(s) := \begin{bmatrix} \frac{1}{\gamma^2}B_1B_1^T & -sI + A & B_2 \\ sI + A^T & C_1^TC_1 & C_1^TD_{12} \\ B_2^T & D_{12}^TC_1 & D_{12}^TD_{12} \end{bmatrix}, W_{21}(s) := \begin{bmatrix} \frac{1}{\gamma^2}C_1^TC_1 & -sI + A^T & C_2^T \\ sI + A & B_1B_1^T & B_1D_{21}^T \\ C_2 & D_{21}B_1^T & D_{21}D_{21}^T \end{bmatrix}.$$

Let the columns of $\Xi_{12} = \begin{bmatrix} \Phi_{12} \\ X_{12} \\ U_{12} \end{bmatrix}$ and $\Xi_{21} = \begin{bmatrix} \Phi_{21} \\ X_{21} \\ U_{21} \end{bmatrix}$ form bases for the generalized eigenspaces corresponding to the $\mathcal{C}^-$ zeros of $W_{12}(s)$ and $W_{21}(s)$ respectively.

Let s_{12_i} and s_{21_i}, $i \in \{1, \ldots, n\}$ be the zeros of the pencils $-sX_{12} + AX_{12} + B_2U_{12}$ and $-sX_{21}^T + X_{21}^TA + U_{21}^TC_2$ respectively. Let s_{γ_i}, $i \in \{1, \ldots, n\}$ denote the the zeros of the pencil $-sX_{21}^TX_{12} + \gamma^{-2}\Phi_{21}^T\Phi_{12}$.

Lemma 1 ([3]) *Controllers $K_Q(s)$ solving the regular sub-optimal $\mathcal{H}^\infty$ problem for the plant (1) exist if and only if each of the following conditions hold:*

- *Both X_{12} and X_{21} have rank n*
- *$Re[s_{12_i}] < 0$ and $Re[s_{21_i}] < 0$ for every $i \in \{1, \ldots, n\}$*
- *$Re[s_{\gamma_i}] > 0$ for every $i \in \{1, \ldots, n\}$*

If all the conditions hold, all such controllers are parameterized by

$$K_Q(s) := \mathcal{LFT}[K(s), Q(s)] \text{ where } K(s) \overset{ss}{=} \left[\begin{array}{c|cc} -sE_k + A_k & B_{k1} & B_{k2} \\ \hline C_{k1} & D_{k11} & D_{k12} \\ C_{k2} & D_{k21} & D_{k22} \end{array}\right] \text{ is given by}$$

$$-sE_k + A_k := \Xi_{21}^TV(s)\Xi_{12} \tag{3}$$

$$\begin{bmatrix} B_{k1} & B_{k2} \end{bmatrix} := \begin{bmatrix} \Xi_{21}^T \begin{bmatrix} 0 \\ 0 \\ -I_{m_2} \end{bmatrix}, & \Xi_{21}^TV(s) \begin{bmatrix} 0 \\ 0 \\ (D_{12}^TD_{12})^{-\frac{1}{2}} \end{bmatrix} \end{bmatrix} \tag{4}$$

$$\begin{bmatrix} C_{k1} \\ C_{k2} \end{bmatrix} := \begin{bmatrix} \begin{bmatrix} 0 & 0 & I_{r_2} \end{bmatrix} \Xi_{12} \\ \begin{bmatrix} 0 & 0 & -(D_{21}D_{21}^T)^{-\frac{1}{2}} \end{bmatrix} V(s)\Xi_{12} \end{bmatrix} \tag{5}$$

$$\begin{bmatrix} D_{k11} & D_{k12} \\ D_{k21} & D_{k22} \end{bmatrix} := \begin{bmatrix} 0 & (D_{12}^T D_{12})^{-\frac{1}{2}} \\ (D_{21}D_{21}^T)^{-\frac{1}{2}} & -(D_{21}D_{21}^T)^{-\frac{1}{2}} D_{22}(D_{12}^T D_{12})^{-\frac{1}{2}} \end{bmatrix} \tag{6}$$

$$\textit{and} \quad V(s) := \begin{bmatrix} \frac{1}{\gamma^2}\left[sI + A^T\right] & \frac{1}{\gamma^2}C_1^T C_1 & \frac{1}{\gamma^2}C_1^T D_{12} \\ \frac{1}{\gamma^2}B_1 B_1^T & -sI + A & B_2 \\ \frac{1}{\gamma^2}D_{21}B_1^T & C_2 & D_{22} \end{bmatrix} \tag{7}$$

and $Q(s)$ is any $r_2 \times m_2$ internally asymptotically stable auxiliary compensator subject to

$$\| Q(s) \|_\infty < \gamma \tag{8}$$

Define also $T(s) := \mathcal{LFT}[G(s), K(s)] \overset{des}{=} \left[\begin{array}{c|cc} -sE_t + A_t & B_{t1} & B_{t2} \\ \hline C_{t1} & D_{t11} & D_{t12} \\ C_{t2} & D_{t21} & D_{t22} \end{array}\right]$. Note that $T_{y_1u_1}(s) = T_{11}(s) = \mathcal{LFT}[G(s), K_{11}(s)]$. Let us now introduce the following pencils: $M_{12}(s) := \begin{bmatrix} -sI + A & B_2 \\ C_1 & D_{12} \end{bmatrix}$, $M_{21}(s) := \begin{bmatrix} -sI + A^T & C_2^T \\ B_1^T & D_{21}^T \end{bmatrix}$ and $N_{B_2}(s) := \begin{bmatrix} -sI + A & B_2 \\ C_1 & D_{12} \\ C_2 & D_{22} \end{bmatrix}$, $N_{C_2}(s) := \begin{bmatrix} -sI + A^T & C_2^T \\ B_1^T & D_{21}^T \\ B_2^T & D_{22}^T \end{bmatrix}$.

The corresponding pencils for the controller are: $M_{k12}(s) := \begin{bmatrix} -sE_k^T + A_k^T & C_{k1}^T \\ B_{k2}^T & D_{k12}^T \end{bmatrix}$, $M_{k21}(s) := \begin{bmatrix} -sE_k + A_k & B_{k1} \\ C_{k2} & D_{k21} \end{bmatrix}$ and $N_{C_{k1}}(s) := \begin{bmatrix} -sE_k^T + A_k^T & C_{k1}^T \\ B_{k1}^T & D_{k11}^T \\ B_{k2}^T & D_{k12}^T \end{bmatrix}$, $N_{B_{k1}}(s) := \begin{bmatrix} -sE_k + A_k & B_{k1} \\ C_{k1} & D_{k11} \\ C_{k2} & D_{k21} \end{bmatrix}$. We note that using the terminology of [16], the zeros of $M_{12}(s)$ and $M_{21}(s)$ are the *invariant zeros* of $G_{12}(s)$ and $G_{21}^T(s)$ respectively.

We may then present the following extension of $\mathcal{H}^\infty$ pole-zero cancellation results of [21, 15] from [9]:

Lemma 2 ([9]) *For every $\gamma > 0$ such that the $\mathcal{H}^\infty$ controller $K(s)$ exists for the plant $G(s)$, and for every choice of $Q(s)$,*

O1. *If $s_o \in \mathcal{C}^-$ is a zero of $M_{12}(s)$ then s_o is a zero of $M_{k21}(s)$ and also an unobservable mode of $T(s)$ and of $T_{y_1u_1}(s)$.*

O2. *If $s_o \in \mathcal{C}^-$ is an unobservable mode of (A, C_1) then s_o is a zero of $N_{B_{k1}}(s)$, a transmission zero of $K_Q(s)$, and also an unobservable mode of $T(s)$ and of $T_{y_1u_1}(s)$.*

O3. *If $s_o \in \mathcal{C}^-$ is a zero of $N_{B_2}(s)$ then s_o is unobservable mode of $(-sE_k + A_k, C_{k2})$, a pole of $K_Q(s)$, and also an unobservable mode of $T(s)$ and of $T_{y_1u_1}(s)$.*

O4. *If $s_o \in \mathcal{C}^-$ is an unobservable mode of $(A, \begin{bmatrix} C_1 \\ C_2 \end{bmatrix})$ then s_o is an unobservable mode of $K(s)$.*

C1. *If $s_o \in \mathcal{C}^-$ is a zero of $M_{21}(s)$ then s_o is a zero of $M_{k12}(s)$ and also an uncontrollable mode of $T(s)$ and of $T_{y_1u_1}(s)$.*

C2. *If $s_o \in \mathcal{C}^-$ is an uncontrollable mode of (A, B_1) then s_o is a zero of $N_{C_{k1}}(s)$, a transmission zero of $K_Q(s)$, and also an uncontrollable mode of $T(s)$ and of $T_{y_1u_1}(s)$.*

C3. *If $s_o \in \mathcal{C}^-$ is a zero of $N_{C_2}(s)$ then s_o is uncontrollable mode of $(-sE_k + A_k, B_{k2})$, a pole of $K_Q(s)$, and also an uncontrollable mode of $T(s)$ and of $T_{y_1u_1}(s)$.*

C4. *If $s_o \in \mathcal{C}^-$ is an uncontrollable mode of $(A, \begin{bmatrix} B_1 & B_2 \end{bmatrix})$ then s_o is an uncontrollable mode $K(s)$.*

While the fact that the $\mathcal{C}^-$ zeros of $M_{12}(s)$ and $M_{21}(s)$ will end up us uncontrollable or unobservable modes of $T_{y_1u_1}(s)$ has been appreciated since at least [15, 21], the *additional* structure obtained in Lemma 2 with regard to the unobservable or uncontrollable modes of (A, B_1, C_1) and the transmission zeros of $\begin{bmatrix} G_{12}(s) \\ G_{22}(s) \end{bmatrix}$ or $\begin{bmatrix} G_{21}^T(s) \\ G_{22}^T(s) \end{bmatrix}$ is significant because in the mixed sensitivity or S/KS synthesis $\mathcal{H}^\infty$ design framework, this is exactly what occurs. For example, the mixed sensitivity framework, with plant $P(s)$, and weights $W_1(s)$, $W_2(s)$ and $W_3(s)$, gives an augmented plant $G(s) = \left[\begin{array}{c|c} W_1(s) & -W_1(s)P(s) \\ 0 & W_2(s) \\ 0 & W_3(s)P(s) \\ \hline I & -P(s) \end{array}\right]$, so that clearly, the poles of $P(s)$ will not be controllable from B_1. Further, if we set $W_2(s) = 0$, then the transmission zeros of $P(s)$ will be the zeros of $N_{B_2}(s)$.

Disconcertingly, Lemma 2 also states that the $\mathcal{C}^-$ zeros of $M_{12}(s)$ or $M_{21}(s)$ will always become unobservable or uncontrollable modes of the closed loop, for all choices of $Q(s)$ and all value of γ for which a solution exists. This motivates the following definition:

Definition 2 *The s_o which are $\mathcal{C}^-$ zeros of either $M_{12}(s)$ or $M_{21}(s)$ will be termed the* **γ-invariant poles** *of the 2-Riccati solution.*

The rest of this paper shows how, in spite of Lemma 2, the pole-zero cancellations between the controller and the plant may be eliminated by proper choice of $Q(s)$.

4 Conversion into an Interpolation Problem

We previously assumed that the plant is such that (A, B_2, C_2) is stabilizable and detectable and this precludes $G(s)$ from being non-minimal in $\bar{\mathcal{C}}^+$. Items **O4**, **C4** of Lemma 2 indicate that the case where the plant is non-minimal in $\mathcal{C}^-$ is of no interest. We may therefore adopt the following assumption without loss of generality:

Assumption 5 *$G(s)$ and $Q(s)$ are minimal.*

We now note that if the plant $G(s)$ is such that at least one of items **O2**, **O3**, **C2**, **C3** of Lemma 2 apply, then we will clearly have cancellations between the poles (or zeros) of controller and the zeros (or poles) of the plant. However the case where only item **O1** (or **C1**) of Lemma 2 applies, no specific pole-zero cancellation actually occurs between the controller and the plant, even though every $\mathcal{C}^-$ zero of $M_{12}(s)$ $(M_{21}(s))$ still emerges as an unobserservable (uncontrollable) mode of the closed loop. We restrict ourselves to considering only the case where items **O2**, **O3**, **C2**, **C3** of Lemma 2 hold:

Assumption 6 *If $s_o \in \mathcal{C}^-$ is a γ-invariant pole of the 2-Riccati solution, then at least one of items* **O2**, **O3**, **C2**, **C3** *of Lemma 2 hold for s_o.*

Assumption 6, in effect, excludes the case where only item **O1** (or **C1**) of Lemma 2 applies, because no specific pole-zero cancellation actually occurs between the controller and the plant. Since, as is is clear from our discussion in the previous section, Assumption 6 includes the cases arising from the mixed sensitivity design framework, it only incurs a very small loss of generality.

Suppose s_o is a $\mathcal{C}^-$ zero of $N_{B_2}(s)$. Then, by Lemma 2, there will be a plant-zero / controller-pole cencellation at s_o. An obvious way to remove closed loop γ-invariant unobservability at s_o is to find $K_Q(s)$ which does not have a pole at s_o. This option is apparently not open to us by Lemma 2. However, we hypothesize that there may exist auxiliary compensators $Q(s)$ which are such that $K_Q(s)$ will have uncontrollable or unobservable modes at the γ-invariant poles of $K(s)$ so that a *minimal realization* of $K_Q(s)$ would not have any pole at s_o.

We now derive necesary and sufficient conditions for $Q(s)$ such that $K_Q(s)$ will have uncontrollable or unobservable modes at specified points on $\mathcal{C}^-$.

Theorem 1 *For all $\gamma > 0$ such that $K(s)$ of Lemma 1 exists,*

- *s_{oR} is an unobservable pole of $(-sE_{K_Q} + A_{K_Q}, C_{K_Q})$ if and only if there exists an $x_{oR} \neq 0$ such that*

$$(-s_{oR}E_k + A_k - B_{k2}D_{k12}^{-1}C_{k1})x_{oR} = 0 \tag{9}$$

  If such an $x_{oR} \neq 0$ exists, and if $Q(s)$ is analytic at s_{oR}, then $Q(s)$ is such that

$$Q(s_{oR})v_{Ro} = w_{Ro} \tag{10}$$

$$\textit{where} \quad v_{oR} := (C_{k2} - D_{k22}D_{k12}^{-1}C_{k1})x_{oR} \tag{11}$$

$$w_{oR} := -D_{k12}^{-1}C_{k1}x_{oR} \tag{12}$$

- *s_{oL} is an uncontrollable pole of $(-sE_{K_Q} + A_{K_Q}, B_{K_Q})$ if and only if there exists an $x_{oL} \neq 0$ such that*

$$x_{oL}^*(-s_{oL}E_k + A_k - B_{k1}D_{k21}^{-1}C_{k2}) = 0 \tag{13}$$

  If such an $x_{oL} \neq 0$ exists, and if $Q(s)$ is analytic at s_{oL}, then $Q(s)$ is such that

$$v_{oL}^*Q(s_{oL}) = w_{oL}^* \tag{14}$$

$$\textit{where} \quad v_{oL}^* := x_{oL}^*(B_{k2} - B_{k1}D_{k21}^{-1}D_{k22}) \tag{15}$$

$$w_{oL}^* := -x_{oL}^*B_{k1}D_{k21}^{-1} \tag{16}$$

Proof: Let the auxiliary compensator be $Q(s) \overset{des}{=} \left[\begin{array}{c|c} -sE_Q + A_Q & B_Q \\ \hline C_Q & D_Q \end{array}\right]$. Since $K_Q(s) := \mathcal{LFT}[K(s), Q(s)] \overset{des}{=} \left[\begin{array}{c|c} -sE_{K_Q} + A_{K_Q} & B_{K_Q} \\ \hline C_{K_Q} & D_{K_Q} \end{array}\right]$, then, keeping in mind $D_{k11} = 0$ and assuming D_Q is such that $\check{D}_{K_Q} := (I - D_QD_{k22})^{-1}$ and $\hat{D}_{K_Q} := (I - D_{k22}D_Q)^{-1}$ exist, we have

$$\begin{bmatrix} -sE_{K_Q} + A_{K_Q} \\ C_{K_Q} \end{bmatrix} = \left[\begin{array}{cc} -sE_k + A_k + B_{k2}\check{D}_{K_Q}D_QC_{k2} & B_{k2}\check{D}_{K_Q}C_Q \\ B_Q\hat{D}_{K_Q}C_{k2} & -sE_Q + A_Q + B_Q\hat{D}_{K_Q}D_{k22}C_Q \\ \hline C_{k1} + D_{k12}D_Q\hat{D}_{K_Q}C_{k2} & D_{k12}\check{D}_{K_Q}C_Q \end{array}\right] \tag{17}$$

($\Rightarrow$). Suppose $(-sE_{K_Q} + A_{K_Q}, C_{K_Q})$ is unobservable at s_{oL}, then there must exist vectors x_{oR}, z_{oR}, at least one non-zero, such that $\begin{bmatrix} -s_{oR}E_{K_Q} + A_{K_Q} \\ C_{K_Q} \end{bmatrix}\begin{bmatrix} x_{oR} \\ z_{oR} \end{bmatrix} = 0$ or equivalently, from (17),

$$(-s_{oR}E_k + A_k + B_{k2}\check{D}_{K_Q}D_QC_{k2})x_{oR} + B_{k2}\check{D}_{K_Q}C_Qz_{oR} = 0 \tag{18}$$

$$B_Q\hat{D}_{K_Q}C_{k2}x_{oR} + (-s_{oR}E_Q + A_Q + B_QD_{k22}\check{D}_{K_Q}C_Q)z_{oR} = 0 \tag{19}$$

$$(C_{k1} + D_{k12}\check{D}_{K_Q}D_QC_{k2})x_{oR} + D_{k12}\check{D}_{K_Q}C_Qz_{oR} = 0 \tag{20}$$

where we have used the identities $D_Q\hat{D}_{K_Q} \equiv \breve{D}_{K_Q}D_Q$ and $\hat{D}_{K_Q}D_{k22} \equiv D_{k22}\breve{D}_{K_Q}$.

If $x_{oR} = 0$, then z_{oR} must be non zero. But since D_{k12} and $\breve{D}_{K_Q}$ must, by definition, be square and have full rank, it must hold from (20) that $C_Q z_{oR} = 0$ and hence $(-s_{oR}E_Q + A_Q)z_{oR} = 0$ from (19), which contradicts the minimality of $Q(s)$, Assumption 5. Hence x_{oR} is non zero. Then using (18) and (20) together to eliminate the z_{oR} terms, we get (9), which is a condition independent of the choice of $Q(s)$.

Given such an x_{oR} exists, then (19)$-B_QD_{k22}D_{k12}^{-1}$(20)$=0$ implies

$$\begin{aligned} 0 &= (-s_{oR}E_Q + A_Q)z_{oR} + B_Q(C_{k2} - D_{k22}D_{k12}^{-1}C_{k1})x_{oR} \\ \Leftrightarrow \qquad z_{oR} &= (s_{oR}E_Q - A_Q)^{-1}B_Q(C_{k2} - D_{k22}D_{k12}^{-1}C_{k1})x_{oR} \end{aligned} \tag{21}$$

since we assume $Q(s)$ has no pole at s_{oR}. Also, manipulating (20) gives

$$C_Q z_{oR} + D_Q(C_{k2} - D_{k22}D_{k12}^{-1}C_{k1}))x_{oR} \;=\; -D_{k12}^{-1}C_{k1}x_{oR}$$

where we have used the fact that $D_{k12}\breve{D}_{K_Q}$ is square and has full rank. Substituting for z_{oR} from (21), and noting that $Q(s) = C_Q(s_{oR}E_Q - A_Q)^{-1}B_Q + D_Q$, we obtain (10), (11), (12).

($\Leftarrow$). At some fixed s_{oR} suppose there exists x_{oR} such that (9) holds and $Q(s)$ is such that (10), (11), (12) hold. The preceding algebra then gives (18), (19), (20) which implies that s_{oR} is an unobservable mode of $K_Q(s)$.

The proof of the conditions for uncontrollability of $K_Q(s)$ is similar and is omitted.

Q.E.D.

We recognize that the conditions (10), (11), (12) and (14), (15), (16) are right and left tangential interpolation constraints on $Q(s)$. If the pertinent unobservable or uncontrollable modes lie in $\mathcal{C}^-$, these constraints, together with the infinity norm bound (8) on $Q(s)$, basically give us a shifted $j\omega$-axis Nevalinna-Pick interpolation problem.

We further note that the conditions imposed by Theorem 1 are exactly satisfied when s_o is a γ-invariant pole corresponding to a zero of $N_{C_2}(s)$ or $N_{B_2}(s)$, or an unobservable or uncontrollable mode of (A, B_1, C_1).

Lemma 3 *For all $\gamma > 0$ such that $K(s)$ of Lemma 1 exists, it holds that there exists x_{oR} such that (9) is satisfied if $s_{oR} \in \mathcal{C}^-$ is either a zero of $N_{C_2}(s)$ or an unobservable mode of (A, C_1). It also holds that there exists x_{oL} such that (13) is satisfied if $s_{oL} \in \mathcal{C}^-$ is either a zero of $N_{B_2}(s)$ or an uncontrollable mode of (A, B_1).*

The proof follows obviously from Lemma 2, and is omitted. Without loss of generality, we set $\gamma := 1$ in the sequel.

5 Bitangential Nevalinna-Pick Interpolation

We will now apply results from the theory of bitangential matrix Nevalinna-Pick interpolation, [1] and references therein. The application of scalar and tangential Nevalinna-Pick interpolation results to $\mathcal{H}^\infty$ control theory is not new, cite [25, 10, 14]. The approach presented here is based on the formulae of [1, Chapter 18].

Suppose we wish to eliminate pole/zero cancellations in the region $\Delta_\alpha \cap \mathcal{C}^-$, where for some fixed $\alpha \in \mathcal{R}^+$, Δ_α is as follows:

$$\Delta_\alpha \;:=\; \left\{ s \in \mathcal{C} : s + \alpha \in \bar{\mathcal{C}}^+ \right\} \tag{22}$$

Clearly, $\Delta_\alpha \supset \bar{\mathcal{C}}^+$, and $\Delta_\alpha \cap \mathcal{C}^- \neq \emptyset$. In that region $\Delta_\alpha \cap \mathcal{C}^-$, suppose we have $s_{L_i} \in (\Delta_\alpha \cap \mathcal{C}^-)$, $i \in \{1, \ldots, N_L\}$ which are either simple $\mathcal{C}^-$ zeros of $N_{B_2}(s)$, or simple $\mathcal{C}^-$ unobservable modes of (A, C_1), and, $s_{R_i} \in (\Delta_\alpha \cap \mathcal{C}^-)$, $i \in \{1, \ldots, N_R\}$ which are either simple $\mathcal{C}^-$ zeros of $N_{C_2}(s)$, or simple $\mathcal{C}^-$ uncontrollable modes of (A, B_1), where s_{L_i}, s_{R_i} are distinct.

Given the sets $\left\{s_{L_1}, \ldots, s_{L_{N_L}}\right\}$ and $\left\{s_{R_1}, \ldots, s_{R_{N_R}}\right\}$ and the 2-Riccati central controller $K(s)$, we may then use the equations (13), (15), (16) and (9), (11), (12) to form the sets $\left\{v_{L_1}, \ldots, v_{L_{N_L}}\right\}$, $\left\{w_{L_1}, \ldots, w_{L_{N_L}}\right\}$ and $\left\{v_{R_1}, \ldots, v_{R_{N_R}}\right\}$, $\left\{w_{R_1}, \ldots, w_{R_{N_R}}\right\}$ respectively. Define also $\tilde{s}_{L_i} := s_{L_i} + \alpha \in \bar{\mathcal{C}}^+$ and $\tilde{s}_{R_i} := s_{R_i} + \alpha \in \bar{\mathcal{C}}^+$.

Define the *Pick* matrix $\Lambda := \begin{bmatrix} S_R & S^* \\ S & S_L \end{bmatrix}$, where $S_R := \left[\dfrac{v_{R_i}^* v_{R_j} - w_{R_i}^* w_{R_j},}{\tilde{s}_{R_j} + \tilde{s}_{R_j}^*}\right]_{1 \leq i,j \leq N_R}$

$$S := \left[\frac{w_{L_i}^* v_{R_j} - v_{L_i}^* w_{R_j}}{\tilde{s}_{L_i} - \tilde{s}_{R_j}}\right]_{1 \leq i \leq N_L, 1 \leq j \leq N_R}, \qquad S_L := \left[\frac{v_{L_i}^* v_{L_j} - w_{L_i}^* w_{L_j}}{\tilde{s}_{L_i} + \tilde{s}_{L_j}^*}\right]_{1 \leq i,j \leq N_L}.$$

Further, let $A_L := diag\left(\tilde{s}_{L_1}, \ldots, \tilde{s}_{L_{N_L}}\right)$, $V_L := \left[v_{L_1}, \ldots, v_{L_{N_L}}\right]$, $W_L := \left[w_{L_1}, \ldots, w_{L_{N_L}}\right]$, $A_R := diag\left(\tilde{s}_{R_1}, \ldots, \tilde{s}_{R_{N_R}}\right)$ and $W_R := \left[w_{R_1}, \ldots, w_{R_{N_R}}\right]$. Then [1, Theorem 18.5.1], with some algebraic manipulation to convert the form of the all-solutions parameterization into a linear fractional transformation form, gives

Lemma 4 *A $p \times q$ rational matrix $\tilde{Q}(s)$ analytic in $\bar{\mathcal{C}}^+$ with $\| \tilde{Q}(s) \|_\infty < 1$ such that $\tilde{Q}(\tilde{s}_{R_i}) v_{R_i} = w_{R_i}$ for all $i \in \{1, \ldots, N_R\}$, and $v_{L_i}^* \tilde{Q}(\tilde{s}_{L_i}) = w_{L_i}^*$ for all $i \in \{1, \ldots, N_L\}$, exists if and only if $\Lambda > 0$. In that case, the required set of all $\tilde{Q}(s)$ is given by*

$$\tilde{Q}(s) := \mathcal{LFT}\{\Theta(s), U(s)\} \text{ where } \Theta(s) \overset{ss}{=} \left[\begin{array}{c|cc} -sE_\Theta + A_\Theta & B_{\Theta 1} & B_{\Theta 2} \\ \hline C_{\Theta 1} & 0 & I_p \\ C_{\Theta 2} & I_q & 0 \end{array}\right], \text{ and}$$

$$-sE_\Theta + A_\Theta := \Lambda^{\frac{1}{2}*} \begin{bmatrix} -sI + A_R & 0 \\ 0 & -sI - A_L^* \end{bmatrix} \Lambda^{-\frac{1}{2}} - \Lambda^{-\frac{1}{2}} \begin{bmatrix} V_R^* \\ -W_L^* \end{bmatrix} \begin{bmatrix} V_R & -W_L \end{bmatrix} \Lambda^{-\frac{1}{2}*}$$

$$\begin{bmatrix} B_{\Theta 1} & B_{\Theta 2} \end{bmatrix} := \Lambda^{-\frac{1}{2}} \left[\begin{array}{c|c} V_R^* & -W_R^* \\ -W_L^* & V_L^* \end{array}\right], \quad \begin{bmatrix} C_{\Theta 1} \\ C_{\Theta 2} \end{bmatrix} := \left[\begin{array}{cc} W_R & -V_L \\ \hline -V_R & W_L \end{array}\right] \Lambda^{-\frac{1}{2}*},$$

and $U(s)$ is any rational matrix function analytic in $\bar{\mathcal{C}}^+$ such that $\|U(s)\|_\infty < 1$.

We then have the following result if we define $Q(s) := \tilde{Q}(s + \alpha)$:

Theorem 2 *It holds that $K_Q(s) := \mathcal{LFT}[K(s), Q(s)]$ will have uncontrollable modes at s_{L_i}, $i \in \{1, \ldots, N_L\}$ and unobservable modes at s_{R_i}, $i \in \{1, \ldots, N_R\}$. Denote the minimal form of $K_Q(s)$ by $\bar{K}_Q(s)$ and let $\bar{T}_{y_1 u_1}(s) := \mathcal{LFT}\left[G(s), \bar{K}_Q(s)\right]$. Then the controller $\bar{K}_Q(s)$ has order at most n and $\bar{T}_{y_1 u_1}(s)$ is internally stable with $\| \bar{T}_{y_1 u_1}(s) \|_\infty < 1$.*

Proof: From Lemma 3, for each s_{L_i} and s_{R_i} there exist x_{L_i} such that $x_{L_i}^*(-s_{L_i} E_k + A_k - B_{k1} D_{k21}^{-1} C_{k2}) = 0$ and x_{R_i} such that $(-s_{R_i} E_k + A_k - B_{k2} D_{k12}^{-1} C_{k1}) x_{R_i} = 0$. Hence, if there exists a rational $m_2 \times r_2$ matrix function $\tilde{Q}(s)$ from Lemma 4 then non-minimality of $K_Q(s)$ follows directly from Theorem 1 and the fact that

$$v_{L_i}^* \tilde{Q}(\tilde{s}_{L_i}) = w_{L_i}^* \quad \Leftrightarrow \quad v_{L_i}^* Q(s_{L_i}) = w_{L_i}^*$$
$$\tilde{Q}(\tilde{s}_{R_i}) v_{R_i} = w_{R_i} \quad \Leftrightarrow \quad Q(s_{R_i}) v_{R_i} = w_{R_i}$$

Since $K(s)$ is of order n, $Q(s)$ is of order $N_L + N_R$ and $N_L + N_R$ distinct modes of $K_Q(s)$ will either be uncontrollable or unobservable, $\bar{K}_Q(s)$ must have order at most n.

Further, $\tilde{Q}(s)$ is analytic on $\bar{\mathcal{C}}^+$ and $\| \tilde{Q}(s) \|_\infty < 1$. It follows that $Q(s) := \tilde{Q}(s+\alpha)$ is analytic on $\bar{\mathcal{C}}^+$ and $\| Q(s) \|_\infty < 1$ since $Q(s)$ is then analytic on $\Delta_\alpha \supset \bar{\mathcal{C}}^+$, and $\sup_{s\in\Delta_\alpha} \| Q(s) \| < 1$. The fact that the resulting closed loop $\bar{T}_{y_1u_1}(s)$ is asymptotically stable and has infinity norm less than 1 follows from Lemma 1.

Q.E.D.

It therefore holds that $\bar{T}_{y_1u_1}(s)$ has no γ-invariant uncontrollable or unobservable poles at s_{L_i}, $i \in \{1,\ldots,N_L\}$ and s_{R_i}, $i \in \{1,\ldots,N_R\}$. We have verified our results for various mixed sensitivty sub-optimal $\mathcal{H}^\infty$ synthesis problems with plants having $\mathcal{C}^-$ poles and zeros close to the $j\omega$-axis. In each case, for a suitable choice of α, using Lemma 4 and Theorem 2, we have been able to genereate a $\mathcal{H}^\infty$ controller which does not produce pole-zero cancellations in the region close to the $j\omega$-axis. The resulting closed loops are also markedly more robust to shifts in plant pole/zero locations compared to the closed loops generated using the 2-Riccati $\mathcal{H}^\infty$ central controllers alone.

Since a sub-optimal 2-Riccati $\mathcal{H}^\infty$ controller will approach an optimal $\mathcal{H}^2$ controller asymptotically as $\gamma \rightarrow \infty$, and since all our results are independent of γ, (provided of course that solutions for both the 2-Riccati and the Nevalinna-Pick steps exist), the results also are applicable to $\mathcal{H}^2$ problems, although the resulting "$\mathcal{H}^2$ controller" will be suboptimal in the sense of [4].

We also note that our results have ready application to the singular $\mathcal{H}^\infty/\mathcal{H}^2$ problem. From the results of [3, 8], it is immediately obvious that the main weakness of that solution offerred, i.e. that the controller generated will produce pole-zero cancellations on $\mathcal{C}^{oe}$ in the closed loop, is but a limiting case of the $\mathcal{C}^-$ pole-zero cancellations of Lemma 2. It therefore holds that internally asymptotically stablizing $\mathcal{H}^\infty/\mathcal{H}^2$ controllers may be generated for the singular problem using Lemma 4 and Theorem 2

6 Summary and Conclusion

Two-Riccati $\mathcal{H}^\infty/\mathcal{H}^2$ synthesis generates controllers which produce closed loop $\mathcal{C}^-$ pole/zero cancellations. This is unacceptable when the plant has $\mathcal{C}^-$ poles or zeros near the $j\omega$-axis. In this paper, we show how 2-Riccati $\mathcal{H}^\infty/\mathcal{H}^2$ controllers may be modified so that pole-zero cancellations near the $j\omega$-axis, which would otherwise occur, can be selectively eliminated. This is achieved by applying matrix Nevalinna-Pick tangential interpolation theory to generate an appropriate auxiliary compensator. The resulting controller will have order not greater than the order of the plant and will produce a more robust closed loop compared to the 2-Riccati central controller.

References

[1] J. Ball, I. Gohberg, and L. Rodman. *Interpolation of Rational Matrix Functions.* Birkhauser Verlag, Basel, 1990.

[2] R. Chiang, M. Safonov, K. Madden, and J. Tekawy. A fixed H_∞ controller for a supermaneuverable fighter performing the herbst maneuver. In *Proc. IEEE Conf. on Decision and Control*, New York, December 1990. IEEE. Honolulu, HI.

[3] B. Copeland and M. Safonov. A generalized eigenproblem solution for singular H^2 and H^∞ problems. In C. Leondes, editor, *Control and Dynamic Systems*, volume 50, pages 331–394. Academic Press, New York, 1992.

[4] J. Doyle, K. Glover, P. Khargonekar, and B. Francis. State space solutions to standard H^2 and H^∞ control problems. *IEEE Trans. on Automatic Control*, AC-24(8):731–747, August 1989.

[5] P. Gahinet. A convex parameterization of H^∞ suboptimal controllers. In *Proc. IEEE Conf. on Decision and Control*, pages 937–942, Tuscon, AZ, December 1992. IEEE Press, New York.

[6] F. R. Gantmacher. *Matrix Theory.* New York: Chelsea, 1959.

[7] K. Glover and J. Doyle. State-space formulae for all stabilizing controllers that satisfy an H^∞-norm bound and relations to risk sensitivity. *Systems and Control Letters*, 11:167–172, 1988.

[8] K. Goh and M. Safonov. The $j\omega$-axis eigenstructure of Hamiltonian pencils. In *Proc. IEEE Conf. on Decision and Control*, Tuscon, AZ, December 16-18, 1992. IEEE Press.

[9] K. Goh and M. Safonov. H^∞ control: Inverse free formulae for $D_{11} \neq 0$ and eliminating pole-zero cancellations via interpolation. In *Proc. IEEE Conf. on Decision and Control*, San Antonio, TX, December 15–17, 1993.

[10] H. Kimura. Directional interpolation approach to H_∞ optimization and robust stabilization. *IEEE Trans. on Automatic Control*, AC-32(12):1085–1093, December 1987.

[11] H. Kwakernaak. A polynomial approach to minimax frequency domain optimization of multivariable feedback systems. *Int. J. Control*, 44(1):117–156, 1986.

[12] A. Laub. Schur techniques in invariant imbedding methods for solving two-point boundary value problems. In *Proc. IEEE Conf. on Decision and Control*, Orlando, FL, December 1982. IEEE Press, New York.

[13] F. Lewis. A survey of linear singular systems. *Circuits, Systems, Signal Processes*, 5(1):3–35, 1986.

[14] D. Limebeer and B. Anderson. An interpolation theory approach to H_∞ controller degree bounds. *Linear Algebra and its Applications*, 98:347–386, 1988.

[15] D. Limebeer and G. Halikias. A controller degree bound for H_∞ -optimal control problems of the second kind. *SIAM J. Control and Optimization*, 26(3):646–677, May 1988.

[16] A. G. J. MacFarlane and N. Karcanias. Poles and zeros of linear multivariable systems: A survey of the algebraic, geometric and complex-variable theory. *Int. J. Control*, 24(1):33–74, 1976.

[17] S. O'Young, I. Postlethwaite, and D.-W. Gu. A treatment of $j\omega$-axis model matching transformation zeros in optimal H^2 and H^∞ control design. *IEEE Trans. on Automatic Control*, AC-34(5):551–553, 1989.

[18] M. Safonov. Imaginary-axis zeros in multivariable H^∞ optimal control. In R. F. Curtain, editor, *Modelling, Robustness and Sensitivity Reduction.* Springer-Verlag, New York, 1987.

[19] M. Safonov, A. J. Laub, and G. L. Hartmann. Feeback properties of multivariable systems: The role and use of the return difference matrix. *IEEE Trans. on Automatic Control*, AC-26(1):47–65, February 1981.

[20] M. Safonov, D. J. N. Limebeer, and R. Chiang. Simplifying the H_∞ theory via loop-shifting, matrix-pencil and descriptor concepts. *Int. J. Control*, 50(6):2467–2488, 1989.

[21] J. Sefton and K. Glover. Pole/zero cancellations in the general H_∞ problem with reference to a two-block design. *Systems and Control Letters*, 14:295–306, 1990.

[22] M. Tsai, E. Geddes, and I. Postlethwaite. Pole-zero cancellations and closed-loop properties of an H_∞ mixed sensitivity design problem. In *Proc. American Control Conf.*, Chicago, IL, June 1990. IEEE Press, New York.

[23] M. Tsai, E. Geddes, and I. Postlethwaite. Pole-zero cancellations and closed-loop properties of an H_∞ mixed sensitivity design problem. *Automatica*, 28(3):519–530, 1992.

[24] P. van Dooren. The generalized eigenstructure problem in linear system theory. *IEEE Trans. on Automatic Control*, AC-26(1):111–129, February 1981.

[25] G. Zames and B. A. Francis. Feedback, minimax sensitivity and optimal robustness. *IEEE Trans. on Automatic Control*, AC-28(5):585–600, May 1983.

LOW-ORDER CONTROL DESIGN FOR LMI PROBLEMS USING ALTERNATING PROJECTION METHODS

Karolos M. Grigoriadis and Robert E. Skelton
Space Systems Control Laboratory
Purdue University
W. Lafayette, IN 47907

Abstract

This paper suggests some simple computational techniques for control design, for problems described by Linear Matrix Inequalities (LMIs). In particular, we concentrate on the stabilization and the suboptimal H_∞ output feedback control design problems. These problems can be described by a pair of LMIs and an additional coupling condition. This coupling condition is convex for the full order control design problem, however convexity is lost for the controller design problem of order strictly less that the plant order. We formulate these problems as feasibility problems with matrix set constraints of simple geometry, and we develop analytical expressions for the projection operators onto these sets. For the full order design problem, our alternating projection techniques are guaranteed to converge globally to a feasible solution. However, for the reduced order case, the convergence is guaranteed only locally. Examples illustrate the suggested approach.

1. Introduction

In the past few years, linear matrix inequalities (LMIs) have received considerable attention for control analysis and design [1-5]. The number of control problems which can be formulated as LMI problems is large and continues to grow. Hence, there is an induced increasing interest in numerical solution of LMI problems. These problems are convex problems and standard nonsmooth convex programming (for example the ellipsoid algorithm or the cutting-plane algorithm) can be applied for a solution. In addition, more specialized methods, based on interior point techniques, have been suggested recently for computational solutions [2,6].

In this paper a new computational approach is suggested for control design problems described by LMIs, based on alternating projection techniques. These techniques have been used successfully in the past in image reconstruction problems [7] (see also [12] and the references therein) and they have recently received attention as control design tools in conjunction with the covariance control design problem [8,9]. In an abstract formulation, the idea behind these techniques is the following: Given a family of closed convex sets, the sequence of alternating projections onto these sets converge to a point in the intersection of the family. Several variations of this standard approach have been suggested to solve minimum distance problems with respect to a family of convex sets [10,11] and to accelerate the rate of convergence using information about the direction towards the intersection [13]. It is important to emphasize that it is required that the sets have a simple geometric structure so that, analytical expressions for the projection operators onto these sets can be derived. This fact provides an advantage of these methods with respect to traditional nonsmooth convex optimization methods (e.g. the ellipsoid algorithm or the cutting plane algorithm [14]) since the structure of the constraint sets is taken into account. In the present work we are also interested in constraint sets which are not convex. In this case, convergence of the alternating projection techniques can be guaranteed only locally, i.e. when the initial starting point is in a neighborhood of a feasible solution [15]. Alternating projections for nonconvex constraint sets have also been used in the past in image reconstruction problems with very satisfactory performance [16].

To formulate the controller design problem, we will take advantage of the parametrization of all stabilizing controllers and all suboptimal H_∞ controllers in terms of Linear Matrix Inequalities (LMIs) provided in [5,3,4]. The full order dynamic output feedback control design problem results in convex (but nonsmooth) feasibility or optimization programs. However, the fixed order control design problem (where the order of the controller is *less* than the order of the plant) results in an LMI formulation with an additional nonconvex coupling constraint, and there exist no efficient numerical techniques to solve this problem.

In the present work we propose alternating projection methods to solve both the full order and the reduced order stabilization and suboptimal H_∞ control design problems. We formulate these problems as feasibility problems with matrix set constraints of simple geometry, and we develop analytical expressions for the projection operators onto these sets. The static state feedback and the full order dynamic output feedback control problems result in convex feasibility problems and the alternating projections are guaranteed to converge to a feasible solution. The alternating projection methods seem to provide a simple, effective and easy to implement numerical technique for solution of these problems, to supplement the existing literature on numerical solutions of convex LMI problems. The reduced order control design problem results in a nonconvex feasibility problem of simple geometry and alternating projection methods are also applied for a numerical solution. In this case, global convergence of the algorithm is not guaranteed, but our numerical experience indicates that the algorithm performs very satisfactory. In fact, in the examples we tried with a known lowest order stabilizing controller, the algorithm always provided a minimum order stabilizing controller.

We will use the following notation: A^T will denote the transpose of a matrix A and $A^\perp$ will denote a matrix with the following properties: $\mathcal{N}(A^\perp) = \mathcal{R}(A)$ and $A^\perp A^{\perp T} > 0$ (where $\mathcal{N}$ denotes the null space and $\mathcal{R}$ the range of a matrix). Note that, for a given A, the matrix $A^\perp$ is not unique, but any such choice will be acceptable throughout this paper. Proofs have been omitted.

2. Preliminaries

In this section the LMI formulation of the stabilization and the H_∞ suboptimal control design problem is reviewed from [5,4]. We consider a linear, time-invariant, continuous-time dynamical system with the following state space representation

$$\dot{x}_p = A_p x_p + B_p u + D_p w \tag{2.1a}$$

$$y = C_p x_p + B_y u + D_y w \tag{2.1b}$$

$$z = M_p x_p \qquad + D_z w \tag{2.1c}$$

where $x_p(t) \in \mathbb{R}^{n_p}$ is the plant state vector, $u(t) \in \mathbb{R}^{n_u}$ is the control input vector, $w(t) \in \mathbb{R}^{n_w}$ is the external input vector (which includes plant disturbances and measurement noise), $z(t) \in \mathbb{R}^{n_z}$ is the regulated output vector and $y(t) \in \mathbb{R}^{n_y}$ is the measured output vector. We seek a linear, time-invariant, continuous-time controller of order n_c, with state space representation

$$\dot{x}_c = A_c x_c + B_c z \tag{2.2a}$$

$$u = C_c x_c + D_c z \tag{2.2b}$$

where $x_c(t) \in \mathbb{R}^{n_c}$ is the controller state vector.

The following result provides necessary and sufficient conditions for the existence of a stabilizing controller of a specified order.

Theorem 2.1 [5]. *The following statements are equivalent.*
(a) There exists a stabilizing dynamic output feedback controller of order n_c.
(b) There exist matrices $X > 0$ *and* $Y > 0$ *such that*

$$B_p^{\perp}(XA_p^T + A_pX)B_p^{\perp T} < 0 \tag{2.3a}$$

$$M_p^{T\perp}(YA_p + A_p^TY)M_p^{T\perp T} < 0 \tag{2.3b}$$

$$\begin{bmatrix} X & I \\ I & Y \end{bmatrix} \geq 0 , \quad \text{rank} \begin{bmatrix} X & I \\ I & Y \end{bmatrix} \leq n_p + n_c . \tag{2.3c}$$

In [5], all stabilizing controllers of order n_c, which correspond to a feasible solution (X,Y) of (2.3), are parametrized in terms of a contraction.

Next, we review the necessary and sufficient conditions for the existence of an H_∞ suboptimal controller of a specified order. Given a scalar $\gamma > 0$ we define an H_∞ suboptimal controller to be a stabilizing controller such that the transfer function from w to y has H_∞ norm less that γ. Without loss of generality we assume $\gamma = 1$.

Theorem 3.1 [4]. *The following statements are equivalent.*
(a) There exists an H_∞ *suboptimal controller of order* n_c.
(b) There exist matrices $X > 0$ *and* $Y > 0$ *such that*

$$\begin{bmatrix} B_p \\ B_y \end{bmatrix}^{\perp} \begin{bmatrix} A_pX + XA_p^T + D_pD_p^T & XC_p^T + D_pD_y^T \\ C_pX + D_yD_p^T & D_yD_y^T - I \end{bmatrix} \begin{bmatrix} B_p \\ B_y \end{bmatrix}^{\perp T} < 0 \tag{2.4a}$$

$$\begin{bmatrix} M_p^T \\ D_z^T \end{bmatrix}^{\perp} \begin{bmatrix} YA_p + A_p^TY + C_p^TC_p & YD_p + C_p^TD_y \\ D_p^TY + D_y^TC_p & D_y^TD_y - I \end{bmatrix} \begin{bmatrix} M_p^T \\ D_z^T \end{bmatrix}^{\perp T} < 0 \tag{2.4b}$$

$$\begin{bmatrix} X & I \\ I & Y \end{bmatrix} \geq 0 , \quad \text{rank} \begin{bmatrix} X & I \\ I & Y \end{bmatrix} \leq n_p + n_c . \tag{2.4c}$$

Given a feasible pair (X,Y), all H_∞ suboptimal controllers of order n_c corresponding to this pair are parametrized in [4] in terms of a contractive matrix.

We note that the parametrization of all stabilizing controllers (2.2) can be easily obtained from the parametrization of all H_∞ suboptimal controllers (2.3) when we let the H_∞ bound γ to approach infinity.

As it is pointed out in [4,5], the parametrizations (2.3) and (2.4) are convex for the case of full order controllers, i.e. when $n_c = n_p$. In this case the rank constraints in (2.3c) and (2.4c) are trivially satisfied and the problems are reduced to LMI convex feasibility problems. However, the reduced order controller case $n_c < n_p$ is no longer convex since the rank conditions in (2.3c) and (2.4c) are not convex in the parameter space (X, Y).

3. A Projection Formulation of the LMI Problem

In this section we reformulate the LMI feasibility problems of the previous section as a matrix feasibility problem of a simpler geometry. This allows us to explicitly derive the expressions for the orthogonal projection operators onto the constrained sets. These projections will lead us to alternating projection techniques for the solution of the stabilization and H_∞ suboptimal control design problems.

To begin, let $\mathcal{S}_n$ be the set of real symmetric $n \times n$ matrices equipped with the following norm (Frobenius norm) and inner product

$$\| X \| = [\mathrm{tr}\,(X^2)]^{1/2} = \left[\sum_{i,j=1}^{n} X_{ij}^2 \right]^{1/2}, \quad < X, Y > = \mathrm{tr}\,(XY), \quad X, Y \in \mathcal{S}_n . \tag{3.1}$$

To start our discussion, we consider the following set of real symmetric matrices

$$\mathcal{L} \triangleq \{ X \in \mathcal{S}_n : EXF + F^T X E^T + Q < 0 \} \tag{3.2}$$

where the matrices E, F and $Q \in \mathcal{S}_n$ are given real matrices of compatible dimensions. It can be easily verified that $\mathcal{L}$ is a convex subset of $\mathcal{S}_n$. Note that each one of the LMI constraints (2.3a), (2.3b), (2.4a) and (2.4b) can be written as in (3.2), by appropriate choices of the matrices E, F and Q. For example, the LMI (2.3a) obtains the form in (3.2) dy defining

$$E \triangleq \begin{bmatrix} B_p \\ B_y \end{bmatrix}^{\perp} \begin{bmatrix} A_p \\ C_p \end{bmatrix}, \quad F \triangleq [\, I \;\; 0 \,] \begin{bmatrix} B_p \\ B_y \end{bmatrix}^{\perp T}, \quad Q \triangleq \begin{bmatrix} B_p \\ B_y \end{bmatrix}^{\perp} \begin{bmatrix} D_p D_p^T & D_p D_y^T \\ D_y D_p^T & D_y D_y^T - I \end{bmatrix} \begin{bmatrix} B_p \\ B_y \end{bmatrix}^{\perp T} .$$

For computational purposes, we prefer to have closed sets (i.e. sets which include their limit points), so we consider the following closed "ε-approximation" of the set $\mathcal{L}$

$$\mathcal{L}_\varepsilon \triangleq \{ X \in \mathcal{S}_n : EXF + F^T X E^T + Q \le -\varepsilon I \} \tag{3.3}$$

where ε is a small positive constant. Hence, the set $\mathcal{L}_\varepsilon$ is closed, convex, it is contained in $\mathcal{L}$ and it approaches $\mathcal{L}$ as $\varepsilon \to 0$. We will use $\mathcal{L}_\varepsilon$ to represent (in an ε-approximation sense) anyone of the constraint sets defined by the LMIs (2.3a), (2.3b), (2.4a) and (2.4b), by an appropriate choice of the matrices E, F, Q and X.

The next result, decomposes the set $\mathcal{L}_\varepsilon$ to two closed, convex sets of simpler geometry by increasing the dimension of the parameter space.

Lemma 3.1. *Define the following sets in* $\mathcal{S}_{2n}$

$$\mathcal{J}_\varepsilon \triangleq \{ W \in \mathcal{S}_{2n} : [\, E \;\; F^T \,] W \begin{bmatrix} E^T \\ F \end{bmatrix} \le -Q_\varepsilon \} \tag{3.4a}$$

$$\mathcal{T} \triangleq \{ W \in \mathcal{S}_{2n} : W = \begin{bmatrix} W_{11} & W_{12} \\ W_{12}^T & W_{22} \end{bmatrix}, W_{11} = W_{22} = 0,\; W_{12} = W_{21} \in \mathcal{S}_n \} \tag{3.4b}$$

where $Q_\varepsilon \triangleq Q + \varepsilon I.$ *Then the following statements are equivalent*

(a)

$$X \in \mathcal{L}_\varepsilon \tag{3.5}$$

(b)

$$X = W_{12} = W_{21} \tag{3.6a}$$

where

$$W \in \mathcal{J}_\varepsilon \cap \mathcal{T} . \tag{3.6b}$$

Therefore the two closed convex sets $\mathcal{J}_\varepsilon$ and $\mathcal{T}$ provide an equivalent description of the LMI constraint set $\mathcal{L}_\varepsilon$. The advantage of describing the set $\mathcal{L}_\varepsilon$ in terms of the intersection $\mathcal{J}_\varepsilon \cap \mathcal{T}$, follows from the fact that we can compute explicit expressions for the orthogonal projection operators onto the sets $\mathcal{J}_\varepsilon$ and $\mathcal{T}$. These expressions are provided by the following results.

Lemma 3.2. *Let* $W \in \mathcal{S}_{2n}$. *Consider the singular value decomposition*

$$[\,E \quad F^T\,] = U\,[\,\Sigma \quad 0\,]\,V^T \tag{3.7a}$$

where U *and* V *are orthogonal matrices, and define*

$$\overline{W} \triangleq V^T W V = \begin{bmatrix} \overline{W}_{11} & \overline{W}_{12} \\ \overline{W}_{12}^T & \overline{W}_{22} \end{bmatrix}, \qquad \overline{W}_{11} \in \mathcal{S}_n\,. \tag{3.7b}$$

Consider the eigenvalue-eigenvector decomposition

$$\overline{W}_{11} + \Sigma^{-1} U^T Q_\varepsilon U \Sigma^{-1} = L \Lambda L^T \tag{3.7c}$$

where Λ *is a diagonal matrix which contains the eigenvalues of the matrix* $\overline{W}_{11} + \Sigma^{-1} U^T Q_\varepsilon U \Sigma^{-1}$ *and* L *is the corresponding orthogonal matrix of the normalized eigenvectors. The projection* $W^* = P_{\mathcal{J}_\varepsilon}(W)$ *of* W *onto the set* $\mathcal{J}_\varepsilon$*, defined in (3.4a), is given by*

$$W^* = V \begin{bmatrix} \overline{W}_{11}^* & \overline{W}_{12} \\ \overline{W}_{12}^T & \overline{W}_{22} \end{bmatrix} V^T \tag{3.7d}$$

where

$$\overline{W}_{11}^* = L \Lambda_- L^T - \Sigma^{-1} U^T Q_\varepsilon U \Sigma^{-1} \tag{3.7e}$$

and Λ_- *is the the diagonal matrix obtained by replacing the positive eigenvalues of* Λ *by zero.*

For a proof of this result see [9].

Lemma 3.3. *Let* $W \in \mathcal{S}_{2n}$*. The orthogonal projection* $W^* = P_{\mathcal{T}}(W)$ *of* W *onto the set* $\mathcal{T}$*, defined by (3.4b), is provided by*

$$W^* = \begin{bmatrix} 0 & X^* \\ X^* & 0 \end{bmatrix} \tag{3.8a}$$

where X^* *is defined by*

$$X^* = \frac{1}{2}(W_{12} + W_{12}^T)\,. \tag{3.8b}$$

In addition to the LMI constraints sets, we are interested to derive explicit expressions for the orthogonal projection onto the positivity and rank constraint sets corresponding to the conditions (2.3c) and (2.4c). To this end, define the following sets in $\mathcal{S}_{2n}$

$$\mathcal{D} \triangleq \{ Z \in \mathcal{S}_{2n} : Z = \begin{bmatrix} X & 0 \\ 0 & Y \end{bmatrix}, X, Y \in \mathcal{S}_n \} \tag{3.9}$$

$$\mathcal{P} \triangleq \{ Z \in \mathcal{S}_{2n} : Z \geq -J \} \tag{3.10}$$

$$\mathcal{R} \triangleq \{ Z \in \mathcal{S}_{2n} : \operatorname{rank}(Z + J) \leq k \} \tag{3.11}$$

where k is a given integer such that $n \leq k \leq 2n$, and

$$J \triangleq \begin{bmatrix} 0 & I_n \\ I_n & 0 \end{bmatrix} \in \mathcal{S}_{2n}\,.$$

Notice that the sets $\mathcal{D}$ and $\mathcal{P}$ are closed convex sets. The expressions for the orthogonal projections onto these sets are provided next.

Lemma 3.4 *Let*

$$Z = \begin{bmatrix} Z_{11} & Z_{12} \\ Z_{12}^T & Z_{22} \end{bmatrix} \in \mathcal{S}_{2n} .$$

The orthogonal projection, $Z^* = P_{\mathcal{D}}(Z)$ of Z onto the set $\mathcal{D}$ is given by

$$Z^* = \begin{bmatrix} Z_{11} & 0 \\ 0 & Z_{22} \end{bmatrix} . \tag{3.12}$$

The orthogonal projection onto the set $\mathcal{P}$ is provided by the following result.

Lemma 3.5 [18]. *Let $Z \in \mathcal{S}_n$ and let $Z + J = L\Lambda L^T$ be the eigenvalue-eigenvector decomposition of $Z + J$, where Λ is the diagonal matrix of the eigenvalues and L is the orthogonal matrix of the normalized eigenvectors. The orthogonal projection $Z^* = P_{\mathcal{P}}(Z)$ of Z onto the set $\mathcal{P}$ is given by*

$$Z^* = L\Lambda_+ L^T - J \tag{3.13}$$

where Λ_+ the diagonal matrix obtained by replacing the negative eigenvalues of Λ by zero.

Hence, this projection requires an eigenvalue-eigenvector decomposition of the $n \times n$ symmetric matrix $Z + J$.

We note that the rank constraint set $\mathcal{R}$, defined by (3.11), is a closed set, but it is not convex. Therefore, given a matrix Z in $\mathcal{S}_{2n}$, there might be several matrices in $\mathcal{R}$ which minimize the distance from Z. We will call any such matrix, a projection of Z on $\mathcal{R}$. The following result provides a projection onto the set $\mathcal{R}$.

Lemma 3.6 [19]. *Let $Z \in \mathcal{S}_{2n}$ and let $Z + J = U\Sigma V^T$, a singular value decomposition of $Z + J$. A projection, $Z^* = P_{\mathcal{R}}(Z)$ of Z onto $\mathcal{R}$ is given by*

$$Z^* = U\Sigma_k V^T - J \tag{3.14}$$

where Σ_k is the diagonal matrix obtained by replacing the $2n - k$ *smallest singular values of $Z + J$ by zero.*

To summarize our results, in this section we reformulated the constraint sets of the the stabilization and H_∞ suboptimal control design problems of section 2, as matrix constraint sets of simpler geometric structure. This enabled us to obtain analytical expressions for the projection operators onto these sets. In the next section we describe alternating projection algorithms which, using analytical expressions for projection operators, can provide solutions to feasibility problems.

4. Alternating Projection Methods

Consider a finite dimensional Hilbert space $\mathcal{H}$ and let $\|\cdot\|$ be the norm induced by the inner product $<\cdot, \cdot>$. In an abstract formulation, the problem we are interested to solve is the following:

Feasibility Problem: Given a family of closed sets $\mathcal{C}_1, \mathcal{C}_2, ..., \mathcal{C}_m$ in $\mathcal{H}$, find a vector $x^* \in \mathcal{H}$ such that

$$x^* \in \mathcal{C} \triangleq \mathcal{C}_1 \cap \mathcal{C}_2 \cap ... \cap \mathcal{C}_m \tag{4.1}$$

or determine that no such vector exists (i.e., the intersection $\mathcal{C}$ is empty).

In this section we will review the alternating projection techniques in [13,15,17] to compute a solution for the feasibility problem (4.1). We begin by considering the case where all sets

C_i are convex.

Let P_i be the orthogonal projection from $\mathcal{H}$ onto the convex set C_i. Recall that P_i is the possibly nonlinear operator defined by $P_i x = \hat{x}$ where x is in $\mathcal{H}$ and $\hat{x}$ is the unique element in C_i solving the following optimization problem

$$\| x - \hat{x} \| = \inf \{ \| x - y \| : y \in C_i \} . \tag{4.2}$$

Now define the composition operator Φ on $\mathcal{H}$ by $\Phi \triangleq P_m P_{m-1} \ldots P_1$. It follows that Φ is a contraction operator and the convex set C is precisely the set of fixed points of C, see [17]. In fact, [17,13] provides an algorithm to compute a vector x in C when C is not empty.

Theorem 4.1 [17,13]. *Let x_o be any vector in $\mathcal{H}$. If C is not empty, then $\Phi^i x_o$ converges to a vector x in C as $i \to \infty$. If C is empty, the sequence $\{x_i\}_{i=0}^{\infty}$ of alternating projections $x_1 = P_1 x_0$, $x_2 = P_2 x_1$,..., $x_m = P_m x_{m-1}$, $x_{m+1} = P_1 x_m$, ... does not converge.*

Therefore, the alternating convex projection algorithm, provided by Theorem 4.1 can solve the feasibility problem (4.1) when the constraint sets C_i are convex. This algorithm can be implemented very easily in a computer, but may suffer from slow convergence (for example, consider the case where C_1 and C_2 are hyperplanes with "small" angles between them). One way to overcome this problem is to make use of some "directional information" in the algorithm. For simplicity of presentation we will state this algorithm for two convex sets, that is $C = C_1 \cap C_2$. The general case is discussed in [13].

Theorem 4.2 [13]. *Assume that $C = C_1 \cap C_2$ is not empty. Let x_o be any vector in $\mathcal{H}$ and define the sequence $\{x_i\}_{i=0}^{\infty}$ by*

$$\left.\begin{aligned} & x_1 = P_1 x_o \,,\; x_2 = P_2 x_1 \,,\; x_3 = P_1 x_2 \\ & x_4 = x_1 + \lambda_1 (x_3 - x_1) \,,\; \lambda_1 = \frac{\| x_1 - x_2 \|^2}{< x_1 - x_3 , x_1 - x_2 >} \end{aligned}\right\} \text{1st cycle}$$

$$\left.\begin{aligned} & x_5 = P_1 x_4 \,,\; x_6 = P_2 x_5 \,,\; x_7 = P_1 x_6 \\ & x_8 = x_5 + \lambda_2 (x_7 - x_5) \,,\; \lambda_2 = \frac{\| x_5 - x_6 \|^2}{< x_5 - x_7 , x_5 - x_6 >} \end{aligned}\right\} \text{2nd cycle} \tag{4.3}$$

$$\vdots$$

Then the sequence $\{x_i\}_{i=0}^{\infty}$ converges to a vector x in C. If C is empty then the sequence $\{x_i\}_{i=0}^{\infty}$ does not converge.

Therefore the algorithm of Theorem 4.2 may provide an efficient computational technique to solve the feasibility problem corresponding to (4.1), for the case where all sets C_i are convex.

Since our control design problems involve a non convex set, we are interested to seek a solution to the feasibility problem (4.1) for the case where some of the sets C_i are not convex. To extend our projection techniques to this case, we define the orthogonal projection to a non-convex set C_i to be given by any solution of the minimum distance problem (4.2). We notice that a given point in $\mathcal{H}$ might have several projection (possibly infinite) to a non convex set C_i. Unfortunately, when some of the sets C_i are non convex, the alternating projection algorithms of Theorems 4.1 and 4.2 are not guaranteed to converge to a feasible solution, even when such a solution exists. However, it can be shown (see [15]) that convergence is still guaranteed in a local sense, i.e. when the starting point of the algorithm is in a neighborhood of a feasible solution. We refer to [15] for more details.

We conclude this section by summarizing the fact that alternating projection techniques might provide simple and easy to implement algorithms for solution of feasibility problems. These algorithms require expressions for the projection operators onto the constraint sets, therefore these sets need to have a simple geometry so that such expressions can be analytically derived. For convex feasibility problems these algorithms converge globally, however for non convex problems the convergence is guaranteed only locally.

5. Numerical Examples

In this section we use the alternating convex projection techniques for control design for the 2-mass-spring system shown in Fig. 1.

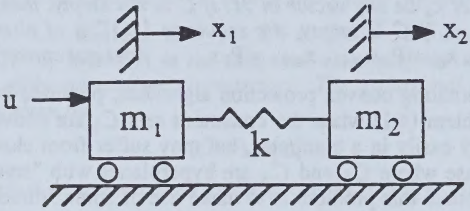


Fig. 1

We assume that the two bodies have equal mass $m_1 = m_2 = 1$ and they are connected by a spring with stiffness $k = 1$. We assume that only the position of body 2 is measured and a control force acts on body 1, i.e the problem is non-collocated.

A state space representation of the system is

$$\dot{x} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} u$$

$$z = [\,0 \;\; 1 \;\; 0 \;\; 0\,]x \qquad (5.1)$$

where x_1 and x_3 are the position and velocity, respectively, of body 1 and x_2 and x_4 are the position and velocity of body 2.

We are interested to design a stabilizing controller for this system, which places the closed loop poles to the left of a vertical line $\mathrm{Re}(z) = -\alpha$ in the complex plane. To this end, we seek a feasible solution to the set of matrix inequalities (2.3), where A_p is replaced by the system matrix $A + \alpha I$, where A is the system matrix of the system in (5.1). We set $\alpha = 0.2$, and we begin with the search of a full order controller ($n_c = 4$). The feasibility problem (2.3) is a convex problem and our alternating projection methods are guaranteed to converge to a feasible solution. Using the directional alternating projection algorithm of Theorem 3.2, along with the orthogonal projection expressions developed in section 2, we obtain the following feasible solution of (2.3).

$$X = \begin{bmatrix} 2.4864 & 1.3467 & -0.4973 & -1.1690 \\ 1.3467 & 3.4575 & 0.6304 & -0.6915 \\ -0.4973 & 0.6304 & 2.9937 & -0.6721 \\ -1.1690 & -0.6915 & -0.6721 & 2.3874 \end{bmatrix}, \; Y = \begin{bmatrix} 2.3874 & -0.6721 & -0.6915 & -1.1690 \\ -0.6721 & 2.9937 & 0.6304 & -0.4973 \\ -0.6915 & 0.6304 & 3.4575 & 1.3467 \\ -1.1690 & -0.4973 & 1.3467 & 2.4864 \end{bmatrix}.$$

It can be easily verified that these matrices satisfy the conditions (2.3). The directional alternating projection algorithm needed 606 iterations to converge to this solution (this corresponds to a CPU time of 152.66 sec). Using the results of [5] we can obtain the following stabilizing controller which corresponds to the feasible set (X, Y)

$$\dot{x}_c = \begin{bmatrix} -0.5231 & 4.6017 & 7.5167 & 0.0000 \\ 0.7067 & -0.4077 & 1.0140 & 0.0000 \\ -0.3783 & -5.5355 & -5.7237 & 0.0000 \\ 0.0000 & 0.0000 & 0.0000 & -33.5333 \end{bmatrix} x_c + \begin{bmatrix} -8.4938 \\ 0.8348 \\ 9.0487 \\ 0.0000 \end{bmatrix} z$$

$$u = [\, -0.2128 \;\; 3.4035 \;\; 4.4224 \;\; 0.0000 \,] x_c + [\, -6.2692 \,] z \,.$$

This controller provides the following closed loop poles

$$-0.2028 \pm 0.4625i \,,\; -0.2257 \pm 1.1387i \,,\; -0.2982 \pm 1.6737i \,,\; -5.2012 \,,\; -33.5333 \,,$$

which obviously satisfy the required settling time constraint.

In the following, we seek for a second order controller ($n_c = 2$) to satisfy the same requirements. In this case we include the rank condition (2.3c) into the problem as discussed in section 3. The directional alternating projection algorithm converged to the following feasible solution.

$$X = \begin{bmatrix} 2.7374 & 1.9351 & -0.5475 & -1.4226 \\ 1.9351 & 4.3499 & 0.6486 & -0.8700 \\ -0.5475 & 0.6468 & 1.5012 & -0.2332 \\ -1.4226 & -0.8700 & -0.2332 & 2.7629 \end{bmatrix} ,\; Y = \begin{bmatrix} 2.7629 & -0.2332 & -0.8700 & -1.4226 \\ -0.2332 & 1.5012 & 0.6486 & -0.5475 \\ -0.8700 & 0.6486 & 4.3499 & 1.9351 \\ -1.4226 & -0.5475 & 1.9351 & 2.7374 \end{bmatrix} .$$

It can be easily checked that these matrices satisfy the conditions (2.3) for $n_c = 2$. In this case the directional alternating projection method required 901 iterations (corresponding CPU time 202.76 sec) for convergence. A second order controller which correspond to the feasible solution (X, Y) is

$$\dot{x}_c = \begin{bmatrix} -0.5667 & -1.6954 \\ 0.6145 & -0.6373 \end{bmatrix} x_c + \begin{bmatrix} 1.2966 \\ 1.3876 \end{bmatrix} z$$

$$u = [\, -0.2271 \;\; -0.5598 \,] x_c + [\, 1.1671 \,] z \,.$$

This controller provides the following closed loop poles

$$-0.2006 \pm 0.3778i \,,\; -0.2007 \pm 0.9358i \,,\; -0.2007 \pm 1.3343i \,,$$

which satisfy the desired pole regional constraint. We note that a second order controller is the lowest order stabilizing controller for the plant (5.1).

6. Conclusions

Alternating convex projection methods are suggested for solution of the stabilization and the H_∞ suboptimal control design problems, described by LMIs. To this end, we formulate these problems as matrix set constraint feasibility problems of simpler geometry and we develop the expressions for the orthogonal projections onto these constraint sets. Alternating projections appear to be effective and simple to implement and they provide an additional tool to the already existing literature on numerical solutions for LMI problems. These techniques are also applied to the reduced order control design problem, although this problem is nonconvex and global convergence of the algorithms is not guaranteed. However, our numerical experience indicates that alternating projection techniques perform very satisfactory even in the case of nonconvex constraint problems.

References

[1] A. Packard, K. Zhou, P. Pandey and G. Becker, "A collection of robust control problems leading to LMIs," Proc. 30th Conf. Dec. Cont., Brighton, England, 1991.

[2] S. Boyd, L. El Ghaoui, E. Feron and V. Balakrishnan, *Linear Matrix Inequalities in Systems and Control Theory*, Monograph in preparation.

[3] P. Gahinet, "A New Representation of H_∞ Suboptimal Controllers," Proc. 1992 Amer. Contr. Conf., Chicago, Illinois, 1992.

[4] T. Iwasaki and R. E. Skelton, "All Controllers for the General H_∞ Control Design Problem: LMI existence Conditions and State Space Formulas,", Proc. 1993 Amer. Contr. Conf., San Francisco, 1993.

[5] T. Iwasaki and R. E. Skelton, "Parametrization of All Stabilizing Controllers via Quadratic Lyapunov Functions," Submitted to J. Optim. Theory Appl.

[6] L. Vandenberghe and S. Boyd, "Primal-Dual potential reduction methods for problems involving matrix inequalities," preprint.

[7] D. C. Youla, and H. Webb, "Image restoration by the method of convex projections: Part 1 - Theory," IEEE Trans. Medical Imag., Vol. 1, No. 2, pp. 81-94, 1982.

[8] K. M. Grigoriadis and R. E. Skelton, "Alternating Convex Projection Methods for Covariance Control Design," Proc. 30th Allerton Conf. Commun. Contr. Comp., Monticello, Illinois, 1992. Also, International Journal of Control, to appear.

[9] K. M. Grigoriadis, R. E. Skelton and A. E. Frazho, "Alternating Convex Projection Methods for Discrete-Time Covariance Control Design," Proc. 32th Conf. Dec. Cont., San Antonio, Texas, 1993.

[10] J. P. Boyle and R. L. Dykstra, "A Method for Finding Projections onto the Intersection of Convex Sets in Hilbert Space," Lecture Notes in Statistics, Vol. 37, Springer-Verlag, pp. 28-47, 1986.

[11] S. P. Han, "A Successive Projection Method," Math. Program., Vol. 40, pp. 1-14, 1988.

[12] L. C. Potter, "Convergence of Successive Approximations for Nonexpansive Operators," Proc. 29th Allerton Conf. Commun. Contr. Comp., pp. 149-158, 1991.

[13] L. G. Gubin, B. T. Polyak and E. V. Raik, "The Method of Projections for Finding the Common Point of Convex Sets," USSR Comp. Math. Phys., Vol. 7, pp. 1-24, 1967.

[14] S. P. Boyd and G. H. Barratt, *Linear Controller Design: Limits of Performance*, Prentice-Hall, New Jersey, 1991.

[15] P. L. Combettes and H. J. Trussell, "Method of Successive Projections for Finding a Common Point of Sets in Metric Spaces," J. Optim. Theory Appl., Vol. 67, No. 3, pp. 487-507, 1990.

[16] C. A. Weber and J. P. Allebach, "Reconstruction of Frequency-Offset Fourier Data by Alternating Projection onto Constraint Sets," Proc. 24th Allerton Conf. Commun. Contr. Comp., pp. 194-201, 1986.

[17] W. Cheney and A. A. Goldstein, "Proximity maps for convex sets," Proc. Amer. Math. Soc., Vol. 12, pp. 448-450, 1959.

[18] N. J. Higham, "Computing the Nearest Symmetric Positive Semidefinite Matrix," Lin. Algebra Appl., Vol. 103, pp. 103-118, 1988.

[19] R. A. Horn and C. R. Johnson, *Matrix Analysis*, Cambridge University Press, 1985.

Numerical Issues in the Computation of $\mathcal{H}^\infty$ Controllers for Distributed Plants [1]

Onur Toker and *Hitay Özbay*
Department of Electrical Engineering
The Ohio State University
Columbus, OH 43210

Abstract

In this paper two block $\mathcal{H}^\infty$ optimal and suboptimal control problems are considered for SISO distributed plants with finitely many unstable modes. A relatively simple expression of the controller is presented. This controller formula is given in terms of finite dimensional spectral factor of rational weighting functions, and two polynomials of degrees n_1+l (n_1+l-1 in the optimal case), where n_1 is the dimension of the sensitivity weight, and l is the number of unstable modes. These polynomials are determined uniquely from $2(n_1+l+1)$ ($2(n_1+l)$ in the optimal case) interpolation conditions, which can be written directly from the controller structure. A delay system example is presented to illustrate the numerical properties of the system of linear equations, which determines the controller.

Notation

$\mathbb{R}$, (resp. $\mathbb{C}$)	The set of real (resp. complex) numbers
$\mathbb{C}_+$, $\overline{\mathbb{C}}_+$	$\{s \in \mathbb{C} \,:\, \text{Re}\,(s) > 0\}$, $\{s \in \mathbb{C} \,:\, \text{Re}\,(s) \geq 0\}$
$\mathbb{C}_\sigma$	$\{s \in \mathbb{C} : \text{Re}\,(s) > -\sigma\}$,
$\mathcal{H}^\infty$, $\mathcal{H}^2$, (resp. $\mathcal{H}^\infty(\mathbb{C}_\sigma)$, $\mathcal{H}^2(\mathbb{C}_\sigma)$),	Hardy spaces defined on $\mathbb{C}_+$, (resp. on $\mathbb{C}_\sigma$)
$\mathcal{B}$	$\{u \,:\, u \in \mathcal{H}^\infty \,,\ \|u\|_\infty \leq 1\}$
$deg(R)$	degree of the polynomial $R(s)$.

1 Introduction

The purpose of this paper is to discuss certain numerical issues involved with the computation of $\mathcal{H}^\infty$ optimal and suboptimal controllers for a class of infinite dimensional SISO, LTI plants with finitely many unstable poles, and rational weighting functions. Recently, we have obtained a simple expression for the $\mathcal{H}^\infty$ controllers, which is given in terms of (i) a spectral factor of the weights (ii) inner outer parts of the plant, and (iii) two unknown polynomials to be determined from a finite set of linear equations. These linear equations are equivalent to certain interpolation conditions on the unknown polynomials. The interpolation conditions can be obtained *directly* from the controller formula. In this paper we give a delay system example to illustrate the numerical computations involved in the construction and solution of the system of linear equations, which determine the controller.

The rest of this paper is organized as follows. Two block $\mathcal{H}^\infty$ optimal and suboptimal control problems are defined in Section 2. In the same section the controller formula is presented. Some remarks on the computational issues are made in Section 3. A delay system example is given in Section 4. A brief discussion on inner/outer factorizations can be found in Section 5, and some concluding remarks are made in Section 6.

---

[1] This work was supported in part by NSF under grant No. MSS-9203418, and by AFOSR under grant No. F49620-93-1-0288

2 $\mathcal{H}^\infty$ Optimal and Suboptimal Controllers

In this paper the plant and the controller are represented by their transfer functions $P(s)$ and $C(s)$, respectively. A system, whose transfer function is $G(s)$, will be said to be stable if $G \in \mathcal{H}^\infty$. The closed loop system is stable (or "C stabilizes P") if transfer functions S, T, CS and PS belong to $\mathcal{H}^\infty$, where $S = (1 + PC)^{-1}$ is the sensitivity function and $T = 1 - S$ is the complementary sensitivity function.

In order to avoid the outer factor absorption problem, for which the reader is referred to [1], it will be assumed that the plant does not have any poles on the imaginary axis. If this assumption is not satisfied originally, one can replace $\mathcal{H}^\infty$ by $\mathcal{H}^\infty(\mathbb{C}_\sigma)$ for some small $\sigma > 0$, in the definition of stability. Then, the modified plant, $P(s - \sigma)$, satisfies this assumption, provided $P(s)$ has finitely many poles in $\mathbb{C}_\sigma$, and no poles on $\mathrm{Re}(s) = -\sigma$. With respect to the original stability definition, this situation corresponds to stability with a margin of at least σ. For simplicity it will also be assumed that the poles of $P(s)$ in $\mathbb{C}_+$, denoted by $\alpha_1, \ldots, \alpha_l$, are distinct. So, $P(s)$ can be written as $N_p(s)/m_d(s)$, where $m_d(s) = \prod_{k=1}^{l} \frac{s-\alpha_k}{s+\alpha_k}$, and $N_p \in \mathcal{H}^\infty$, which can be factored as (see e.g. [2]) $N_p = m_n N_o$, where $m_n \in \mathcal{H}^\infty$ is inner (i.e. all-pass function, possibly infinite dimensional) and $N_o \in \mathcal{H}^\infty$ is outer (i.e. minimum phase, possibly infinite dimensional). This factorization is unique up to multiplication by -1. We will also assume that $N_o = N_{o1} N_{o2}$ where $N_{o1}, N_{o1}^{-1}, N_{o2} \in \mathcal{H}^\infty$, and N_{o2} is rational. This factorization is not unique. But it will only be used for concluding that the optimal/suboptimal $\mathcal{H}^\infty$ controller obtained here is proper. Note that if two plants P_1 and P_2 satisfy the above assumptions, then so does their product $P_1 P_2$.

Given weighting functions W_1 and W_2, the optimal mixed sensitivity problem is to find

$$\gamma_o := \inf_{C \text{ stabilizes } P} \left\| \begin{bmatrix} W_1 S \\ W_2 T \end{bmatrix} \right\|_\infty \tag{1}$$

and the optimal $\mathcal{H}^\infty$ controller, denoted by C_{opt}, corresponding to the optimal performance level γ_o. That is, C_{opt} stabilizes the plant P, and yields

$$\left\| \begin{bmatrix} W_1(1 + PC_{opt})^{-1} \\ W_2 PC_{opt}(1 + PC_{opt})^{-1} \end{bmatrix} \right\|_\infty = \gamma_o.$$

The suboptimal mixed sensitivity problem is to parametrize the set

$$\mathcal{C}_\rho = \left\{ C \ : \ C \text{ stabilizes } P, \ \left\| \begin{bmatrix} W_1 S \\ W_2 T \end{bmatrix} \right\|_\infty \leq \rho \right\} \tag{2}$$

for a given desired performance level $\rho > \gamma_o$.

In this paper we assume that $W_1(s)$ and $W_2(s)$ are rational functions. For properness of the optimal controller it will be assumed that $W_1(s)$ is non-constant and W_1, $(W_2 N_{o2})$, and $(W_2 N_{o2})^{-1}$ are in $\mathcal{H}^\infty$, see [4] for a detailed discussion on these assumptions, and relations to other types of two block $\mathcal{H}^\infty$ control problems.

We now summarize the results of [6] on the computation of γ_o, C_{opt} and $\mathcal{C}_\rho$.

Let $\eta_1, \ldots, \eta_{n_1} \in \overline{\mathbb{C}}_+$, be the poles of $W_1(-s)$ (if η_i has multiplicity ℓ_i then it is assumed to be repeated ℓ_i times in this list), and set

$$E_\rho(s) := \left(\frac{W_1(-s) W_1(s)}{\rho^2} - 1 \right). \tag{3}$$

The zeros of $E_\rho(s)$ are denoted by $\beta_1, \ldots, \beta_{2n_1}$, and they are assumed to be distinct. Then, β_i's can be enumerated in such a way that $\beta_1, \ldots, \beta_{n_1}$ are in $\overline{\mathbb{C}}_+$, and $\beta_{n_1+i} = -\beta_i$. We also assume that $m_n(\beta_k) \neq 0$ for $k = 1, ..., 2n_1$, and $\beta_k \neq \alpha_j$ for $k = 1, .., 2n_1$, $j = 1, .., l$. Now define

$$F_\rho(s) := G_\rho(s) \prod_{k=1}^{n_1} \frac{s - \eta_k}{s + \eta_k} \tag{4}$$

where $G_\rho \in \mathcal{H}^\infty$ is minimum phase and determined from the spectral factorization

$$G_\rho(s)G_\rho(-s) := \left(1 - (\frac{W_2(-s)W_2(s)}{\rho^2} - 1)\ E_\rho(s)\right)^{-1} . \tag{5}$$

Then (under certain genericity assumptions see [6] for complete details) the optimal $\mathcal{H}^\infty$ controller is given by

$$C_{opt}(s) = E_{\gamma_o}(s)m_d(s)\ \frac{N_o(s)^{-1}F_{\gamma_o}(s)L(s)}{1 + m_n(s)F_{\gamma_o}(s)L(s)} \tag{6}$$

where $L(s) = L_2(s)/L_1(s)$, and $L_1(s), L_2(s)$ are polynomials of degrees less than or equal to $(n_1 + l - 1)$ and they satisfy

$$\begin{aligned}
0 &= L_1(\beta_k) + m_n(\beta_k)F_{\gamma_o}(\beta_k)L_2(\beta_k) & k = 1, \ldots, n_1 \qquad (7)\\
0 &= L_1(\alpha_k) + m_n(\alpha_k)F_{\gamma_o}(\alpha_k)L_2(\alpha_k) & k = 1, \ldots, l \qquad (8)\\
0 &= L_2(-\beta_k) + m_n(\beta_k)F_{\gamma_o}(\beta_k)L_1(-\beta_k) & k = 1, \ldots, n_1 \qquad (9)\\
0 &= L_2(-\alpha_k) + m_n(\alpha_k)F_{\gamma_o}(\alpha_k)L_1(-\alpha_k) & k = 1, \ldots, l. \qquad (10)
\end{aligned}$$

Note that (7–8) correspond to interpolation conditions that the denominator term $(1 + m_n(s)\ F_{\gamma_o}(s)\ L(s))$ must cancel the closed right half plane zeros of $E_{\gamma_o}(s)m_d(s)$. This means, in particular, that $m_d(s)$ term in the numerator of $C_{opt}(s)$ does not cancel the unstable poles of the plant. Moreover, since $W_1, (W_2N_o)^{-1} \in \mathcal{H}^\infty$, the term $F_{\gamma_o}N_o^{-1}$ is proper. Also note that (7–10) constitute $2(n_1 + l)$ linear homogenous equations in $2(n_1 + l)$ unknown coefficients of $L_1(s)$ and $L_2(s)$. If γ_o is replaced by a variable, say γ, in equations (7–10), then a new set of linear homogenous equations is obtained, in terms of $2(n_1 + l)$ unknown coefficients, for each fixed γ. In [6] it was shown that, under certain genericity assumptions, γ_o is the largest value of γ for which there is a non-trivial solution to these $2(n_1 + l)$ linear homogenous equations. That is, γ_o can be found by plotting smallest singular values of the matrix representation of these equations, as γ varies in an interval. The largest value of γ for which the plot shows a zero is γ_o, see the example below.

The set of all suboptimal $\mathcal{H}^\infty$ controllers are given by $\mathcal{C}_\rho = \{C_U(s) \ : \ U \in \mathcal{B}\}$ with

$$C_U(s) = E_\rho(s)m_d(s)\ \frac{N_o(s)^{-1}F_\rho(s)L_U(s)}{1 + m_n(s)F_\rho(s)L_U(s)} \tag{11}$$

where

$$L_U(s) = \frac{L_2(s) + L_1(-s)U(s)}{L_1(s) + L_2(-s)U(s)} \quad , \quad U \in \mathcal{B},$$

and $L_1(s), L_2(s)$ are polynomials of degree $\leq n_1 + l$ satisfying (7-10) with γ_o replaced by ρ, and the following two conditions:

$$
\begin{aligned}
0 &= L_2(-a) + (E_\rho(a) + 1)F_\rho(a)m_n(a)L_1(-a) && (12)\\
1 &= L_1(-a), && (13)
\end{aligned}
$$

where a is an arbitrary positive real number not equal to any of the α_i's and β_j's. For different values of a one can obtain different parametrizations of the suboptimal controllers; see Section 4 for a detailed discussion and an example. Also, (13) can be replaced by $L_1(-a) \neq 0$, because L_U depends only on the ratios of these polynomials.

3 On the Optimal $\mathcal{H}^\infty$ Controller

In the optimal case L_1 and L_2 are in the form $L_i(s) = L_{i0} + L_{i1}s + \cdots + L_{i(n_1+l-1)}s^{n_1+l-1}$ for $i = 1, 2$. By defining $\Psi_i = [L_{i0} \quad L_{i1} \quad \cdots \quad L_{i(n_1+l-1)}]^T$ it is possible to re-write equations (7–10) as follows:

$$
M_\gamma \begin{bmatrix} \Psi_1 \\ \Psi_2 \end{bmatrix} = 0 \qquad \text{where} \qquad M_\gamma = \begin{bmatrix} \mathrm{V}_+ & \mathrm{DV}_+ \\ \mathrm{DV}_- & \mathrm{V}_- \end{bmatrix} \tag{14}
$$

$$
\mathrm{V}_- = \mathrm{V}_+\mathrm{J} \quad , \quad \mathrm{V}_+ = \begin{bmatrix} 1 & \beta_1 & \cdots & \beta_1^{n_1+l-1} \\ \vdots & \vdots & \cdots & \vdots \\ 1 & \beta_{n_1} & \cdots & \beta_{n_1}^{n_1+l-1} \\ 1 & \alpha_1 & \cdots & \alpha_1^{n_1+l-1} \\ \vdots & \vdots & \cdots & \vdots \\ 1 & \alpha_l & \cdots & \alpha_l^{n_1+l-1} \end{bmatrix},
$$

J and D are diagonal matrices whose ith diagonal entries are $(-1)^{i-1}$ and $m_n(\beta_i)F_\gamma(\beta_i)$ ($m_n(\alpha_{i-n_1})F_\gamma(\alpha_{i-n_1})$, if $i \geq n_1 + 1$), respectively, $i = 1, \ldots, n_1 + l$. One of the genericity assumptions, is that V_+ is invertible (that is α_i's and β_j's are distinct). Then, it is possible to solve Ψ_1 uniquely, in terms of Ψ_2, and vice versa:

$$
\Psi_1 = -\hat{\mathrm{D}}\Psi_2 \qquad \text{where} \qquad \hat{\mathrm{D}} = \mathrm{V}_+^{-1}\mathrm{DV}_+. \tag{15}
$$

Using (15) in (14) the number of linear equations to be solved reduces to $n_1 + l$:

$$
(\mathrm{I} - (\mathrm{J}\hat{\mathrm{D}})(\mathrm{J}\hat{\mathrm{D}}))\Psi_2 = 0. \tag{16}
$$

Then, from (15,16) it is easy to see that Ψ_1 and Ψ_2 must satisfy

$$
\begin{aligned}
\Psi_2 &= \kappa \, \mathrm{J} \, \Psi_1 && (17)\\
0 &= (\mathrm{I} - \kappa \, \hat{\mathrm{D}} \, \mathrm{J}) \, \Psi_1, && (18)
\end{aligned}
$$

where, either $\kappa = +1$, or $\kappa = -1$. Therefore, one of the eigenvalues of $\hat{\mathrm{D}}\mathrm{J}$ must be either $+1$ or -1, and Ψ_1 must be the corresponding eigenvector. It is also interesting to note that using (17) one can re-write $L(s) = L_2(s)/L_1(s)$ as

$$
L(s) = \kappa \, \frac{[1 \quad s \quad \cdots \quad s^{n_1+l-1}] \, \mathrm{J} \, \Psi_1}{[1 \quad s \quad \cdots \quad s^{n_1+l-1}] \, \Psi_1}. \tag{19}
$$

Hence, in the optimal case $L(s)$ is an all-pass (not necessarily stable) transfer function.

4 Example

We now consider the following plant

$$P(s) = \frac{e^{-hs}(s-\sqrt{2})}{(s+1)(s+0.1-e^{-h_1 s})}, \qquad h_1 = 2\ln(5/3), \qquad h > 0.$$

The only point in $\overline{\mathbb{C}}_+$ where $(s+0.1-e^{-h_1 s})$ becomes zero is $s = 0.5$, (see section 6). So, the plant has only one pole in $\mathbb{C}_+$. It is also easy to verify that the multiplicity of this pole is one. Therefore, in this example inner/outer factors of P can be identified as follows:

$$\begin{aligned} m_n(s) &= e^{-hs}\frac{(s-\sqrt{2})}{(s+\sqrt{2})} & \qquad m_d(s) &= \frac{s-0.5}{s+0.5} \\ N_{o1}(s) &= \frac{(s-0.5)(s+\sqrt{2})}{(s+0.1-e^{-h_1 s})(s+0.5)} & \qquad N_{o2}(s) &= \frac{1}{s+1}. \end{aligned}$$

Suppose that the weights are given as $W_1(s) = (s+1.41)/(s+1.1)$, $W_2(s) = s+1.52$ and the amount of time delay is $h = 0.2$.

In order to find C_{opt}, one needs to compute γ_o and the corresponding all-pass function $L(s)$. It is possible to obtain an upper and lower bound for γ_o, see [5]. For example, using the formulae given in [6], we obtain $1 < \gamma_o < 16.65$. We construct M_γ for sufficiently large number of points in the range $1 < \gamma < 16.65$, to get a meaningful plot of $\sigma_{min}(M_\gamma)$. This plot showed that for $\gamma \geq 7$, M_γ is nonsingular. So, it is sufficient to concentrate in the region $1 < \gamma < 7$. Figure 1 shows the plot $\sigma_{min}(M_\gamma)$ in this region, where solid line is for $h = 0.2$ and the dotted line is for $h = 0.5$. We see that, when $h = 0.2$, for $\gamma = 1.35$ and $\gamma = 5.532$, M_γ is singular. This figure also shows that the zero at $\gamma = 1.3502$ remains fixed after we increase h. At this point, at least one of the genericity conditions is violated. The largest zero $\gamma = 5.5320$ moves to the right with increasing h; and the genericity conditions are not violated at that point. In conclusion for $h = 0.2$ we have $\gamma_o = 5.532$.

In this analysis we took 100 equally spaced γ points between 1 and 7. Although this curve is smooth enough, we don't know in general how may γ points should be taken in order not to miss a zero between two consecutive γ points. General rule of thumb is to start with a small number of points, say 10 or 20, then observe if the curve is smooth, otherwise increase the number of points until a smooth curve is obtained.

Once we find $\gamma_o = 5.5320$, which makes M_{γ_o} singular, we can compute $L(s)$ as described above. Then, $L(s)$ and $F_{\gamma_o}(s)$ determine the optimal controller. For this example we found that

$$L(s) = \frac{s+1.0902}{s-1.0902} \qquad F_{\gamma_o}(s) = \frac{s^2-1.21}{0.1778s^3+0.7258s^2+0.9723s+0.4280}$$

The solid line in Figure 2 shows the magnitude of C_{opt}.

In order to analyze the suboptimal case, let $\rho = 5.7 > \gamma_o$ and choose the parameter a as 2.34. In this case, by using (7–10,12,13) we obtain

$$L_1(s) = 1.1161s^2 + 1.1029s - 2.5308 \quad , \quad L_2(s) = 0.9847s^2 + 3.3796s + 2.5146$$

and

$$F_\rho(s) = \frac{s^2-1.21}{0.1727s^3+0.7051s^2+0.9445s+0.4158}.$$

Figure 2 shows the magnitudes of C_U for the free parameter $U(s) = 0$ (dashed line) and $U(s) = 1$ (dotted line). Note that, in this case C_U is very close the optimal controller C_{opt}, for both $U(s) = 0$ and $U(s) = 1$.

In order to see what happens when ρ is considerably larger than γ_o we now let $\rho = 7$ and $a = 1.34$ (which is arbitrary). In this case,

$$L_1(s) = 0.9513s^2 - 0.1767s - 0.9449 \quad , \quad L_2(s) = 0.5823s^2 + 1.4153s + 0.8509$$

$$F_\rho(s) = \frac{s^2 - 1.21}{0.1414s^3 + 0.5771s^2 + 0.7731s + 0.3403}.$$

Figure 3 shows the magnitudes of C_U for the free parameter $U(s) = 0$ (dashed line) and $U(s) = 1$ (dotted line). Note that, in this case for different values of $U \in \mathcal{B}$, the controllers that we obtain are significantly different from the optimal controller C_{opt} (solid line).

An alternative expression of $L_U(s)$ can be obtained by defining $L_0(s) = L_2(s)/L_1(s)$ and $D(s) = L_2(-s)/L_1(s)$:

$$L_U(s) = L_0(s)\frac{1 + U(s)/D(-s)}{1 + D(s)U(s)}.$$

For the above numerical examples we have

Case 1: $\rho = 5.7$ and $a = 2.34$

$$L_0(s) = 0.8823\frac{(s + 2.3415)(s + 1.0906)}{(s + 2.0788)(s - 1.0907)} \quad D(s) = 0.8823\frac{(s - 2.3415)(s - 1.0906)}{(s + 2.0788)(s - 1.0907)}$$

Case 2: $\rho = 7$ and $a = 1.34$

$$L_0(s) = 0.6121\frac{(s + 1.3402)(s + 1.0903)}{(s + 0.9081)(s - 1.0938)} \quad D(s) = 0.6121\frac{(s - 1.3402)(s - 1.0903)}{(s + 0.9081)(s - 1.0938)}$$

Numerical examples given above indicate that when ρ is close to γ_o, $L_0(s)$ is close to $L(s)$ of the optimal case. Moreover, the free part of the controller, i.e. the ratio $\frac{1+U(s)/D(-s)}{1+D(s)U(s)}$, is close to 1 for all $U \in \mathcal{B}$. This means that, when ρ is close to γ_o, all suboptimal controllers are "close" to the optimal controller. But when ρ is considerably larger than γ_o, the central controller ($U = 0$ case) is significantly different than the optimal controller. Furthermore, in this case the central controller depends on the choice of a; and the range of $\frac{1+U(s)/D(-s)}{1+U(s)D(s)}$ is considerably larger.

Finally, in order to illustrate the effect of different choices of the parameter a, we solve the suboptimal control problem for $\rho = 7$ using a different a value.

Case 3: $\rho = 7$ and $a = 4.0$

$$L_0(s) = 0.3495\frac{(s + 4.0356)(s + 1.0932)}{(s + 1.4852)(s - 1.0937)} \quad D(s) = 0.3495\frac{(s - 4.0356)(s - 1.0932)}{(s + 1.4852)(s - 1.0937)}$$

Note that $L_0(s)$ and $D(s)$ are different from the $L_0(s)$ and $D(s)$ given in case 2, but still we parametrize the same set of suboptimal controllers.

5 On the Inner Outer Factorizations

As we see from the controller formula given in Section 2, the key step in finding $\mathcal{H}^\infty$ optimal controllers is to compute the inner and outer parts of the plant, which is assumed to be infinite dimensional. For delay systems $P(s) = e^{-hs}R(s)$, where $R(s)$ rational, inner outer factorization is easy. Because, in this case the outer factor is simply the outer factor of $R(s)$ and the time delay appears as a factor of the numerator inner part.

For retarded time delay systems of the form

$$P(s) = e^{-Ts}\frac{q_0(s) + q_1(s)e^{-hs} + \ldots + q_n(s)e^{-nhs}}{r_0(s) + r_1(s)e^{-\tau s} + \ldots + r_m(s)e^{-m\tau s}} \tag{20}$$

(where $\deg(q_0) > \deg(q_i)$, $\deg(r_0) > \deg(r_j)$, for $i = 1, \ldots, n$, $j = 1, \ldots, m$ and $\deg(r_0) > \deg(q_0)$), the inner/outer factorization amounts to finding the $\mathbb{C}_+$ roots of

$$Q(s) = q_0(s) + q_1(s)e^{-hs} + \ldots + q_n(s)e^{-nhs} = 0, \tag{21}$$

and the $\mathbb{C}_+$ roots of the denominator of (20). There are several results on this subject, we refer the reader to [3] for an algorithm, and further references. In the special case $n = 1$ it is quite easey to find all $\mathbb{C}_+$ roots. Following is a summary of the procedure we have used in the above example.

Step 1: Find a sufficiently large $\sigma \geq 0$ such that $Q(s+\sigma)$ has no zero in $\mathbb{C}_+$. For example the Nyquist plot of $Q(s+\sigma)/(s+a)^n$ can be used in this step. Here $a > 0$ and $n \geq \deg(q_0)$. Since there are only finitely many $\mathbb{C}_+$ roots of $Q(s)$, there exists $\sigma_{max} \geq 0$ such that all roots of $Q(s)$ lie in $0 \leq Re(s) \leq \sigma_{max}$.

Step 2: Note that if $s = \sigma + j\omega$ is a root of $Q(s)$, then $j\omega$ is a root of

$$H_\sigma(x) = q_0(\sigma - x)q_0(\sigma + x) - e^{-2h\sigma}q_1(\sigma - x)q_1(\sigma + x).$$

Therefore, we can vary σ from 0 to σ_{max} and check whether $H_\sigma(x)$ has a purely imaginary root. If $j\omega$ is a purely imaginary root of $H_\sigma(x)$, then we check whether $Q(\sigma + j\omega) = 0$. In this way, the problem is reduced to a one dimensional search.

Let us now illustrate this method on the example given in Section 5. For

$$Q(s) = s + 0.1 - e^{-h_1 s} \quad \text{where} \quad h_1 = 2\ln(5/3)$$

define $R_1(s) = Q(s+1)$, and $R_2(s) = R_1(s)/(s+10)$. Figure 4 shows the Nyquist plot of $R_2(s)$.

We see that $Q(s)$ has no zero in $Re(s) \geq 1$. Now we can apply Step 2, as σ varies from 0 to 1, we find roots $x_1, \ldots, x_k$ of $H_\sigma(x)$ and plot $\min_k Q(\sigma + jIm(x_k))$ versus σ, as shown in Figure 5. From these plots, we see that the only $\mathbb{C}_+$ zero of $Q(s)$ is $s = 0.5$.

6 Concluding Remarks

In this paper, we have discussed certain computational aspects of the $\mathcal{H}^\infty$ controller formulae given in [6], for a class of unstable distributed plants, with rational weighting functions. The controllers are in the form (6,11), where m_d, m_n, N_o are inner and outer parts of the plant $P = m_n N_o / m_d$; E_ρ is given in terms of the sensitivity weighting function W_1; F_ρ is computed from a spectral factorization which depends on ρ, W_1 and W_2; and polynomials L_1 and L_2 are computed from a finite set of linear equations. These linear equations can be written *directly* as interpolation conditions on L_1 and L_2, that are expressed in terms of m_n and F_ρ evaluated at the zeros of E_ρ, and at the poles and zeros of m_d. The number of equations is $2(n_1 + l + 1)$ (in the optimal case it is $2(n_1 + l)$), which is equal to the number of unknown coefficients of L_1 and L_2. We gave a delay system example to illustrate the numerical construction of these linear equations, and their solutions.

References

[1] Flamm, D.S., "Outer factor 'Absorption' for H^∞ control problems," ISS Report No.55, Princeton University, 1990.

[2] Hoffman, K., *Banach Spaces of Analytic Functions*, Dover Publications, New York, reprinted from Prentice Hall, 1962.

[3] Manitius, A., H. Tran, G. Payre and R. Roy, "Computation of eigenvalues associated with functional differential equations," *SIAM J. Sci. Stat. Comput.*, **8** (1987), pp. 222–247.

[4] Özbay, H., "$\mathcal{H}^\infty$ Optimal controller design for a class of distributed parameter systems", *Int. Journal of Control*, 1993.

[5] Özbay, H., M. C. Smith and A. Tannenbaum, "Mixed sensitivity optimization for a class of unstable infinite dimensional systems," *Linear Algebra and its Appl.*, **178** (1993), pp. 43-83.

[6] Toker, O., and H. Özbay, "H^∞ optimal and suboptimal controllers for infinite dimensional SISO plants," submitted for publication; a brief version is to appear under the title "H^∞ Controllers for Unstable Distributed Plants" in Proc. of the CDC, San Antonio TX, December 1993.

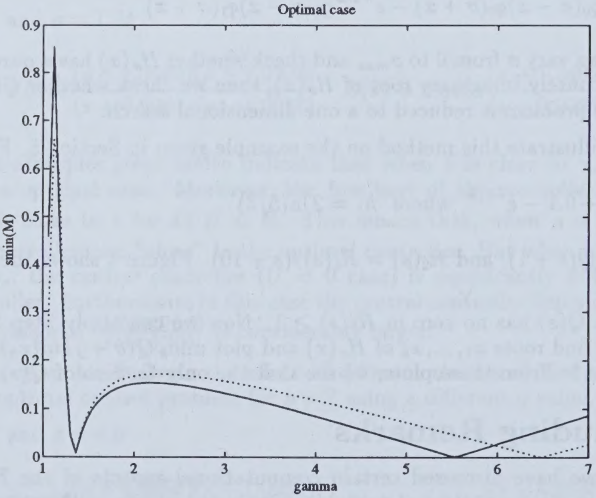


Figure 1: $\sigma_{min}(M_\gamma)$ versus γ plots: $h = 0.2$ solid line, $h = 0.5$ dotted line

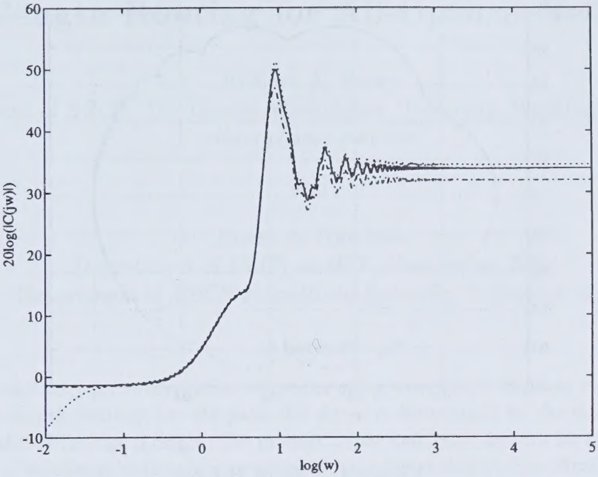


Figure 2: C_{opt}, solid line; $C_{subopt}(U = 0)$, dashed line; $C_{subopt}(U = 1)$, dotted line for $a = 2.34, \rho = 5.7$

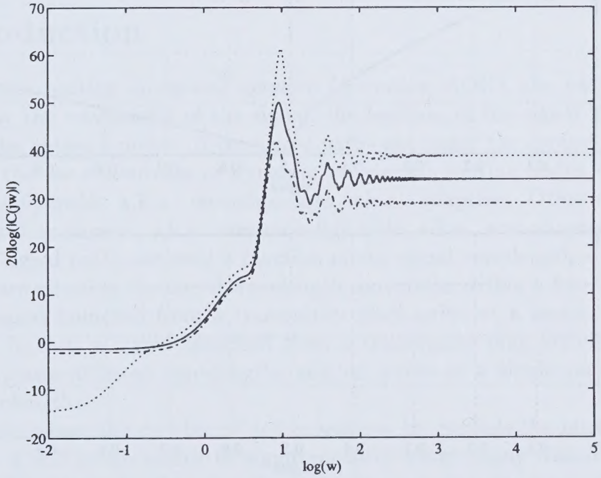


Figure 3: C_{opt}: solid line; $C_{subopt}(U = 0)$: dashed line; $C_{subopt}(U = 1)$: dotted line; for $a = 1.34$, $\rho = 7$

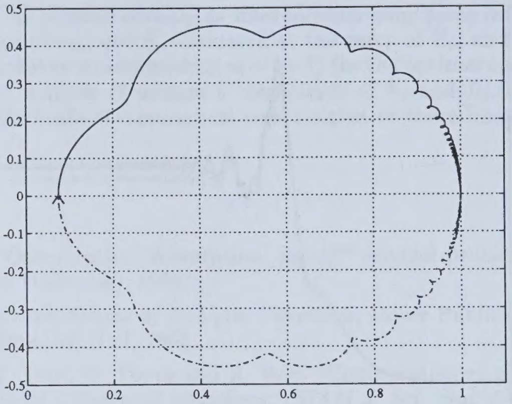


Figure 4: Plot of $R_2(j\omega)$

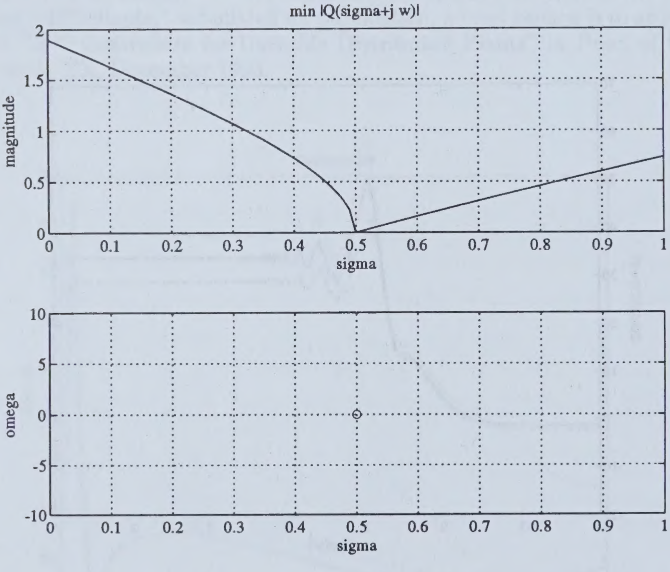


Figure 5: $\mathbb{C}_+$ zeros of $Q(s)$

Wavelength Routing for All-Optical Networks[1]

Richard A. Barry
Departartment of EECS, The George Washington University, Washington DC 20050
ribarry@seas.gwu.edu

and

Pierre A. Humblet
Department of EECS at MIT, Cambridge, MA
Department of EECS at Institute Eurecom, Valbonne, Fr.

Abstract

We consider passive all-optical networks using wavelength division multiplexing and wavelength routing, i.e. the path of a signal is determined by the wavelength of the signal and the signal origin. We present upper and lower bounds on the required number of wavelengths to achieve a given blocking probability. Specifically we show that between $\Omega\left((\sqrt{\rho M})^{(1-P_b)}\right)$ and $O\left(\sqrt{\rho M}\right)$ wavelengths are required to support ρM active session requests in a network with M users and blocking probability P_b. The lower bound holds for all networks with passive wavelength routing and fixed wavelength changing devices. The upper bound is a passive construction without wavelength changers.

Keywords: Wavelength Routing, All-Optical Networks, Connectors.

1 Introduction

In a wavelength routing all-optical network (λ-routing AON), the path of a signal is determined by the wavelength of the signal, the location of the signal transmitter, and the state of the network nodes. If the signal paths are under the control of the network, e.g. through the use of switches or dynamic wavelength routing nodes, we say that the network is *configurable*, a.k.a. reconfigurable, a.k.a. adaptive. Otherwise, we say the network is *fixed* or *passive*, a.k.a. non-reconfigurable, a.k.a. non-adaptive. In a passive network, the signal paths are only a function of the signal wavelength and origin.

Since we are allowing the use of wavelength conversion within a fixed or configurable network, a signal launched from a transmitter may arrive at a receiver on a different wavelength. In fact, a signal launched from a transmitter may arrive at a variety of receivers on many different wavelengths and/or arrive at a single receiver on several different wavelengths.

In networks where the number of active sessions far exceeds the number of available wavelengths, it will be necessary to simultaneously assign many transmitters the same wavelength. Since two signals using the same wavelength cannot travel over the same fiber simultaneously, certain collisions need to be prevented. In particular, we must insure that signals do not collide at any intended receiver. That is, if receiver m is listening to wavelength λ at time t, we must insure that only one signal arrives at receiver m on wavelength λ and time t. If two or more arrive, we say there is *contention.*

---

[1] Research supported by the Army Research Office under contracts DAAL03-86-K-0171 and DAAL03-92-G-0115 and by NSF grant NCR-9206379 and under DARPA grant MDA972-92-J-1038.

Contention is avoided by isolating signals of the same wavelength onto different fibers using λ-routing and λ-changing. Previously, it has been shown that there is a limit to the possible amount of isolation, or equivalently a limit on the wavelength re-use, in non-blocking networks [1, 2]. This limit depends on the number of wavelengths, the number of devices, the functionality of the devices, and the requirements of the users. In section 5, we generalize this result and show that the same limit applies to networks where a small probability of blocking P_b is tolerated. We then present a passive wavelength efficient network with arbitrarily small blocking probability [Section 6]. The limit on wavelength re-use allows the possibility of wavelength changing; the construction does not use any wavelength changing.

First in sections 2 and 3 we formalize the network model and problem statement, respectively. Then a survey of previous relevant results are discussed [section 4]. More comprehensive surveys on wavelength routing can be found in [3, 4, 5].

2 Network Model

We consider networks with M users and F wavelengths. Each user has one fully tunable transceiver[2]; for results on AONs where either the transmitter or the receiver are fixed tuned, see [8]. Each transmitter (receiver) is connected to one outgoing (incoming) fiber.[3] To model wavelength changing, we define an *origin-destination* channel, or OD channel, as an ordered pair of wavelengths $f : f'$. We say that transmitter n is connected to receiver m on OD channel $f : f'$ if a signal launched from n on wavelength f is received at m on wavelength f'. If a transmitter or receiver is tuned to wavelength f, we say that it is *assigned* f. Note that there is no assumed relationship between the OD channels connecting transmitter n to receiver m and the OD channels connecting transmitter m to receiver n.

Using the OD channel terminology, the connectivity of a λ-routing network can be fully described by the set $\mathcal{H} = \{H_\psi \mid \psi \in \Psi\}$, where $H_\psi(n,m)$ is the set of OD channels connecting transmitter n to receiver m in *switching state* ψ and Ψ are the switching states of the network. A switching state represents the joint state of all devices in the network, i.e. switches, wavelength routers, and wavelength changers.

In networks without wavelength changing, $f : f' \in H_\psi(n,m)$ implies that $f = f'$. In this case, we use the obvious short hand notation of f for $f : f'$.

If $|\Psi| = 1$, the network is passive, $\mathcal{H} = \{H\}$ and we use the notation of H for $\mathcal{H}$. A broadcast nework is the simplest example of a passive network where $H(i,j) = \{1, 2, ..., F\}$ for all (i,j).

Also, if $F = 1$, then the network is a conventional circuit switched network. In this case, the number of switches $\log|\Psi|$ required for non-blocking operation is $\Theta(M \log M)$ [10, 11].

---

[2] Since we are not considering multi-point connections, we view the alternate assumption of multiple transceivers per user as a single transceiver and a switch. For results on optical networks with multiple transceivers, see [6, 7].

[3] If transceivers have access to multiple fibers, then some switching mechanism is required to isolate or select the signal on only one of the fibers. Therefore this option is equivalent to a transceiver and a switch. For results on optical networks of this type, see [9].

3 Problem Statement and Previous Results

Define a *session* (n, m) as an ordered pairing of a transmitter to a receiver. We assume that each session requires one full wavelength of bandwidth; for results when sessions require less bandwidth, see [12, 6, 7]. A *traffic* is a set of sessions; we only consider traffics without multi-point connections.

A non-blocking network is one which can support all possible traffics. Non-blocking networks require at least $\sqrt{M/e}$ wavelengths [1].

Most previously work has focused on the asymptotic growth of the required number of wavelengths for non-blocking operation. Rearrangeably and wide-sense non-blocking networks with and without fixed wavelength changers have been considered. Currently there is no known benefit of using wavelength changers and only a small benefit of being rearrangeable. Specifically, it has been shown that no more than $\Theta(\sqrt{M \log M})$ wavelengths are required for non-blocking operation; see [2] for a proof using wavelength changers and [4, 13] for two different proofs without wavelength changers. These results apply to rearrangeable and wide-sense non-blocking networks, but the constant in front of the $\sqrt{M \log M}$ term can be made slightly smaller for the rearrangeable case [14].

Although it is known that there exist networks requiring only $\Theta(\sqrt{M \log M})$ wavelengths, no explicit constructions with this efficiency are known. The best construction without wavelength changing is due to [4] and requires $O\left(2^{(\log M)^{.8+o(1)}}\sqrt{M}\right)$ wavelengths. However $2^{(\log M)^{.8+o(1)}}$ decays so slowly that more wavelengths are needed for this construction than in a broadcast network unless $M \geq 2^{32}$. For reasonable M, the best constructions with and without wavelength changing are $M^{2/3}$ and $\left\lceil \frac{M}{2} + 2 \right\rceil$ respectively [2]. The $M^{2/3}$ construction was obtained by using an analogy to 2-stage switching networks and then using a construction in [15]. The $\left\lceil \frac{M}{2} + 2 \right\rceil$ wavelength network is also the best known strict sense non-blocking network. There is currently no lower bound for the number of wavelengths required in a strict sense non-blocking network, however for the special case of *Light Tree AONs* [16], the last construction is the optimal strict sense non-blocking network [14].

Therefore, although non-blocking passive networks are unscalable, i.e. $M \leq eF^2$, wavelength routing can significantly increase a network's capacity over a broadcast star. This is one reason passive λ-routing networks are being considered as part of a larger configurable AON [16].

4 Problem Statement and New Results

Since the construction of wavelength efficient non-blocking networks has been so elusive and since a small amount of blocking in a network is usually tolerable, we turn our attention and effort to networks with blocking. In this case, we have essentially solved the problem. First we show in section 5 that passive blocking networks are still unscalable. Then in section 6, we show that we can meet the theoretical lower bound with an explicit construction, beating the best existence proof by a factor of $\sqrt{\log M}$.

Consider a one-shot routing problem with ρM random session *requests* $s_1, ..., s_{\rho M}$ where all lists of ρM requests without multi-point connections are equally likely. We call ρ the *utilization* of and ρM the *load* on the network, respectively.

A non-blocking network can always *honor* all requests. A blocking network *blocks* some of the requests. Define the blocking probability P_b to be the expected fraction of blocked requests.

P_b depends on the network and the strategy used for honoring or blocking. We distinguish between two types of strategies: *sequential* and *non-sequential.* In a sequential strategy, the network must honor or block request i without knowledge of requests $i+1, ..., \rho M$. If request i is honored, it must be assigned an OD channel which does not contend with any previously honored requests; we do not allow the possibility of re-arranging OD channels on previously honored requests. The *greedy* strategy is a particularly simple sequential strategy which honors request i iff there exists a feasible OD channel.

In a non-sequential strategy, the network waits until the last request before honoring or blocking any request. A special case of a non-sequential strategy is an *optimal* strategy. An optimal strategy always honors the maximum possible number of requests from any list of requests. Note that the optimal strategy depends on the network and *may* be difficult to determine and/or implement; however the concept is useful for lower bounding the required number of wavelengths. In addition, we will see that for the efficient construction presented in section 6, the optimal strategy is trivial to determine and implement; in fact in this case, the optimal strategy is the greedy strategy.

For example, recall that in a broadcast AON each transmitter is connected to each receiver on all wavelengths. Since there can be at most F honored requests, the optimal and greedy blocking probabilities of a broadcast AON are both $P_b = 1 - \frac{F}{\rho M}$. Therefore $F = (1 - P_b)\rho M = \Theta(\rho M)$ wavelengths are required for a broadcast AON.[4] We will see that far fewer wavelengths are required if wavelength routing is used to spatially isolate signals of the same wavelength on different fibers and that λ-changing is not required to realize this savings in the number of wavelengths.

In the next two sections we present a lower bound on the number of required wavelengths for any passive network possibly using wavelength changing and then explicitly construct passive networks without wavelength changing which come very close to the bound. An optimal strategy is assumed for the lower bounds, a greedy strategy for the constructions. Specifically if $F(M, \rho, P_b)$ is the minimum number of wavelengths required for any passive λ-routing AON with M users, blocking probability no more than P_b, and load at least $\rho M \geq L(P_b)$, we show that

$$\left(\sqrt{\frac{\rho M}{e}}\right)^{1-P_b} \left(1 + O\left(\frac{\ln \rho M}{\rho M}\right)\right) - 1 \;\leq\; F(M, \rho, P_b) \;\leq\; c(P_b)\sqrt{\rho M} \qquad (1)$$

where $c(P_b)$ is a constant which depends only on P_b and is between 5 and 9 for P_b between 10^{-3} and 10^{-6}. The upper bound in eqn. (1) is valid for $\rho M \geq L(P_b)$ where L is a constant which depends only on P_b and increases as P_b decreases. For $P_b = 10^{-6}$, $L \leq 721$ so the fact that $\rho M \geq L$ is a relatively minor restriction.

Notice that except for excessively large P_b, the lower bound is approximately $\sqrt{\rho M/e}$, the same as the non-blocking case; however, unlike the non-blocking case, we have essentially met the theoretical lower bound, i.e. $\Omega\left(\left(\sqrt{\rho M}\right)^{1-P_b}\right) \leq F(M, \rho, P_b) \leq O\left(\sqrt{\rho M}\right)$. Also unlike the non-blocking case, the networks which achieve this wavelength efficiency are explicit constructions (the constructions use a simple generalization of the well known WDM cross-connect [17, 18]).

---

[4] Recall that we are assuming that each session requires a full wavelength of bandwidth.

5 Lower Bound for Blocking Networks

In this section, we prove the lower bound in eqn. (1). A few preliminaries will simplify the discussion.

Recall that a traffic ϕ is a set of sessions. If $(n, m) \in \phi$, we say that (n, m),n, and m are *active* in ϕ. We assume that each transmitter and each receiver are active in at most one session in any traffic, that is we do not consider any multi-point connections. If two sessions are active in the same traffic ϕ, they are said to be *concurrent* in ϕ. We say that a network *supports* ϕ if all the sessions in ϕ can be connected without contention.

Consider any passive AON with connectivity matrix H and let $\mathcal{T}(H)$ be the set of traffics H can support.[5] Also recall that $s_1, s_2, ..., s_{\rho M}$ is a list of session requests and that an optimal strategy is one which always honors the maximal number of requests possible. Formally, an optimal strategy is one which always honors

$$q_\phi = \max\{|\phi'| \quad : \phi' \subseteq \phi \quad \text{and} \quad \phi' \in \mathcal{T}(H)\} \tag{2}$$

sessions where $\phi = \{s_1, ..., s_{\rho M}\}$. Note again that the optimal strategy depends on the network.

The expected number of honored requests under an optimal strategy is

$$E[q] \quad = \quad \frac{1}{\binom{M}{\rho M}^2 (\rho M)!} \sum_\phi q_\phi \tag{3}$$

where the sum is taken over all traffics with ρM sessions. The optimal blocking probability is

$$P_b \quad = \quad \frac{\rho M - E[q]}{\rho M} \tag{4}$$

In this section, we will lower bound P_b by first proving the bound for the special case of $\rho = 1$. Then using the following lemma, we can simply substitute ρM for M in the general case of $\rho < 1$.

Lemma 1 *Let $F(M, \rho, P_b)$ be the minimum number of wavelengths for a network with M users, ρM requests, and blocking probability no more than P_b. Then $F(M, \rho, P_b) \geq F(\rho M, 1, P_b)$.*

Proof. *Consider an arbitrary passive AON operating with an optimal strategy and a blocking probability P_b. Conditioning on the set of transmitters and set of receivers requesting sessions, the blocking probability can be written as the expected blocking probability given a set of transmitters Tran and a set of receivers Rec, i.e.*

$$P_b \quad = \quad \frac{1}{\binom{M}{\rho M}^2} \sum_{Tran, Rec} P_b(Tran, Rec) \tag{5}$$

where the sum is taken over all sets of ρM transmitters and all sets of ρM receivers and where $P_b(Tran, Rec)$ is the blocking probability given Tran and Rec.

---

[5] Since the traffics supported by a network are determined by H, we will informally refer to H as the network itself.

Pick a $(Tran, Rec)$ with $P_b(Tran, Rec) \leq P_b$; there must be at least one. Now form a new network with the ρM transmitters in $Tran$ and the ρM receivers in Rec with the same wavelength connectivity between these users as the original network. Then at least $F(\rho M, 1, P_b(Tran, Rec))$ wavelengths are required for this new network. Since $P_b(Tran, Rec) \leq P_b$, the result follows. □

We need the following lemma the proof of which can be found in [2, 14].

Lemma 2 *A passive λ-routing network can support at most $(F+1)^{2M}$ different traffics, i.e. $|\mathcal{T}(H)| \leq (F+1)^{2M}$ for any H.*

We are now in a position to prove the bound.

Theorem 3 Lower Bound for Networks with Blocking
At least

$$F(M, \rho, P_b) \geq \left(\sqrt{\frac{\rho M}{e}}\right)^{1-P_b} \left(1 + O\left(\frac{\ln \rho M}{\rho M}\right)\right) - 1 \tag{6}$$

wavelengths are needed for any passive AON with blocking probability P_b.

Proof. *Let H be any connectivity matrix and $\mathcal{T} = \mathcal{T}(H)$ be the traffic set the network supports. We use a slight abuse of notation and set $|H| = |\mathcal{T}(H)|$ to be the number of traffics supported by the network. By the lemma, it is sufficient to prove the theorem for $\rho = 1$.*

The first step of the proof is to show that if $E[q]$ is large, then $|H|$ must also be large. Specifically, we first show that for any integer $0 \leq k \leq M-1$,

$$E[q] \leq k + |H| \left(\frac{M}{M-1}\right) \frac{(k+1)}{(M)_{k+1}} \tag{7}$$

where $(n)_i = \binom{n}{i} i!$ is the lower factorial function.

The expected number of honored requests routed under an optimal strategy is

$$E[q] = \sum_{q=1}^{M} qP(q) \tag{8}$$

where $P(q)$ is defined to be the probability that the maximum number of sessions that can be honored is q. Define $P'(q) \geq P(q)$ to be the probability of being able to honor q sessions, i.e. the probability that there exists a q-subset of ϕ (a subset of ϕ with size q) that is supported by the network. So for any k between 1 and $M-1$,

$$E[q] \ \leq k + \Sigma_{q=k+1}^{M} qP'(q) \tag{9}$$

Now let $\mathcal{T}_k$ be the set of traffics in $\mathcal{T}(H)$ with exactly k sessions. There are $\binom{M}{q}$ q-subsets of ϕ and if any of these subsets is in $\mathcal{T}_q$, we can route q sessions. The probability that one of these subsets picked at random is in $\mathcal{T}_q$ is

$$\frac{|\mathcal{T}_q|}{\binom{M}{q}^2 q!} \tag{10}$$

since there are a total of $\binom{M}{q}^2 q!$ possible q-traffics, i.e. traffics of size q, and if we pick a q-subset of ϕ any q-traffic is equally likely. Using the union bound, the probability of being able to honor q requests is no more than

$$P'(q) \;\le\; \frac{\binom{M}{q}|T_q|}{\binom{M}{q}^2 q!} = \frac{|T_q|}{(M)_q}. \tag{11}$$

Therefore, the expected number of honored requests is upper bounded by

$$E[q] \;\le\; k + |H| \sum_{q=k+1}^{M} q\left(\frac{\alpha_q}{(M)_q}\right) \tag{12}$$

where $\alpha_q = |T_q|/|H|$ and $\sum_{q=k+1}^{M} \alpha_q \le 1$. It is not difficult to upper bound the sum, from which it follows that

$$E[q] \;\le\; k + |H|\left(\frac{M}{M-1}\right)\frac{k+1}{(M)_{k+1}} \tag{13}$$

which proves eqn. (7).

For the second part of the proof, pick $k = \lfloor E[q] \rfloor - 1 \le M - 1$. For this k, the second term in eqn. (7) above must be at least 1, and solving for $|H|$ this gives

$$|H| \;\ge\; \left(\frac{M-1}{M}\right)\left(\frac{1}{\lfloor E[q] \rfloor}\right)(M)_{\lfloor E[q] \rfloor} \tag{14}$$

Now $(M)_n \ge \frac{\sqrt{2\pi}}{e}(M/e)^n$ which follows from $\sqrt{2\pi n}(n/e)^n \le n! \le e\sqrt{n}(n/e)^n$ [19]. So,

$$|H| \;\ge\; \left(\frac{M-1}{M}\right)\left(\frac{1}{\lfloor E[q] \rfloor}\right)\left(\frac{\sqrt{2\pi}}{e}\right)\left(\frac{M}{e}\right)^{\lfloor E[q] \rfloor} \tag{15}$$

and since $E[q] \ge \lfloor E[q] \rfloor \ge E[q] - 1$, since $E[q] = M(1 - P_b)$, and since the number of supported traffics is no more than $(F+1)^{2M}$,

$$(F+1) \;\ge\; \epsilon\left(\sqrt{\frac{M}{e}}\right)^{1-P_b} \tag{16}$$

where

$$\epsilon \overset{\text{def}}{=} \left[\left(1 - \frac{1}{M}\right) \cdot \frac{1}{M(1-P_b)} \cdot \frac{\sqrt{2\pi}}{e} \cdot \frac{e}{M}\right]^{\frac{1}{2M}} \tag{17}$$

Now $\ln \epsilon = O\left(\frac{\ln M}{M}\right)$ so that $\epsilon = 1 + O\left(\frac{\ln M}{M}\right)$. In fact, $\epsilon \approx 1$ even for moderate M. For instance, for all $P_b \le .5$ and $M \ge 1000$, $\epsilon \ge .99$. □

6 Efficient Passive Constructions

In the previous section, we essentially showed that at least $\sqrt{\rho M/e}$ wavelengths are required for any network with a small blocking probability. Here we will construct a network with a very small blocking probability using less than $c\sqrt{\rho M}$ wavelengths, where c is a small constant which depends on P_b.

We call the construction a LAN-LR since it consists of N local area networks (LANs) interconnected by a Latin Router (LR) [18]. Specifically, there are N transmitting LANs (*T-LANs*) with $\frac{M}{N}$ transmitters each and N receiving LANs (*R-LANs*), each with $\frac{M}{N}$ receivers.

Numbering the T-LANs from 0 to $N-1$, let t_n be the T-LAN of transmitter $n = 1, 2, \ldots M$. Similarly number the R-LANs and let r_m be the R-LAN of receiver $m = 1, 2, \ldots M$. The connectivity matrix of the LAN-LR is specified by

$$H(n,m) \;=\; \{f \mid \left\lfloor \frac{f}{k} \right\rfloor = t_n - r_m\} \tag{18}$$

k is called the *coarseness* of the network and represents the number of wavelengths connecting any T-LAN to any R-LAN. Using the terminology in [4], a [T-LAN,R-LAN] pair is called a *block*. Note that the total number of wavelengths is $F = Nk$.

Before deriving the blocking probability of a LAN-LR, note that for all $\rho \geq \frac{1}{N}$, the LAN-LR requires M wavelengths to be non-blocking since all transmitters in a T-LAN could request all receivers in an R-LAN. Therefore $k = \frac{M}{N}$ for non-blocking operation and $F = Nk = M$ wavelengths are required. This makes the LAN-LR a very wavelength **inefficient** non-blocking connector even for very small ρ.[6] However, we will see below that for proper choice of N and k, the LAN-LR is a very wavelength **efficient** blocking connector.

To see that, let's derive P_b, the expected fraction of blocked requests. Since there are N^2 blocks and since up to k requests can be honored in any block,

$$P_b \;=\; \frac{1}{\lambda} \sum_{i=k+1}^{M/N} (i-k)P(i) \tag{19}$$

where $P(i)$ is the probability of i requests in a block and $\lambda \overset{\text{def}}{=} \rho M/N^2$ is the expected number of requests per block. It is not too difficult to show that

$$P(i) \;\leq\; P'(i) \overset{\text{def}}{=} \frac{1}{\sqrt{2\pi i}} \left(\frac{\lambda e \gamma^2}{i}\right)^i \tag{20}$$

where $\gamma = exp(k/F)$. Since $P'(i+1) \leq \frac{\gamma^2 \lambda}{k} P'(i)$, we have that

$$P_b \;\leq\; \frac{P'(k+1)}{\lambda} \sum_{j=0}^{b-1} (j+1) \left(\frac{\lambda \gamma^2}{k}\right)^j \tag{21}$$

$$\leq\; \frac{P'(k+1)}{\lambda} \left(1 - \frac{\lambda\gamma^2}{k}\right)^{-2}. \tag{22}$$

Finally, plugging in for $P'(k+1)$ and simplifying,

$$P_b \;\leq\; \left(1 - \frac{\lambda\gamma^2}{k}\right)^{-2} \frac{\gamma^{2k}}{k\sqrt{2\pi k}} \left(\frac{\rho M k e}{F^2}\right)^k \tag{23}$$

---

[6] Recall that $F = \Theta(\sqrt{M \log M})$ wavelength suffice for a non-blocking connector.

We first use this bound to choose a good value of k holding the number of wavelengths F and the total number of users M fixed but allowing the number of LANs N and the number of users per LAN $\frac{M}{N}$ to vary with k. Then using this choice of k, we bound the number of wavelengths required to achieve a desired blocking probability. We will see that for proper choice of k, the blocking probability can be made very small. On the other hand, if care is not taken in choosing k, many more wavelengths may be required [14].

For large F, i.e. $F \geq 2k^2$, the bound on P_b is mainly dependent on the last term. Therefore, we choose k to minimizes that term: $k = k_o \stackrel{\text{def}}{=} F^2/e^2\rho M$. For this value of k,

$$P_b \;\leq\; \left(\frac{e^2}{e^2-\gamma_o}\right)^2 \cdot \frac{\gamma_o^{2k_o}}{\sqrt{2\pi}} \cdot e^{-\{\frac{F^2}{e^2\rho M}+3\ln\frac{F}{e\sqrt{\rho M}}\}} \tag{24}$$

where $\gamma_o \stackrel{\text{def}}{=} exp\{k_o/F\}$. Now if $F \geq 2k_o^2$ or equivalently $\rho M \geq 4k^3/e^2$, $\gamma_o^{2k_o} \leq e$ and $\gamma_o^2 \approx 1$. Assuming that this is so, the following table lists various values of this bound as a function of F. Also listed are the minimum ρM for which the bounds are valid. For smaller ρM, eqn. (23) should be optimized over k considering the effect of γ^{2k}.

F	k	$min\{\rho M\}$	$P_b \leq$
$(e\sqrt{5})\sqrt{\rho M}$	5	68	$.9 * 10^{-3}$
$(e\sqrt{7})\sqrt{\rho M}$	7	185	$.7 * 10^{-4}$
$(e\sqrt{9})\sqrt{\rho M}$	9	395	$.7 * 10^{-5}$
$(e\sqrt{11})\sqrt{\rho M}$	11	721	$.7 * 10^{-6}$

Notice that we did not relax the integer constraints on k. However, we did relax the integer constraints on N and $\frac{M}{N}$. Let's quickly address the validity of these two approximations. Since $N = F/k$, we should have restricted ourselves to k which divide F. But since k divides $F + a$ for some $a < k$, the number of wavelengths can be increased by at most k. Now imagine that we did not relax the integer constraints on $\frac{M}{N} = \frac{Mk}{F}$, but wished to keep the number of users M fixed. We could still use a network with a LR backbone and N LANs, but some LANs will have $\lfloor Mk/F \rfloor$ users and some $\lceil Mk/F \rceil$ users. Since in either case, the number of users is $(Mk/F)(1 + O(1/\sqrt{M}))$, the approximation is sufficient for our purposes.

7 Conclusions

Unlike the non-blocking case, we have been able to construct wavelength efficient blocking networks. Specifically for a blocking probability of P_b, the minimum number of wavelengths is between $\Omega((\sqrt{\rho M})^{1-P_b})$ and $O(\sqrt{\rho M})$. The lower bound allows the possibility of fixed wavelength conversion; the upper bound is achieved by the LAN-LR network which does not use wavelength changing and which has recently been proposed as part of a larger Wide Area AON [16].

References

[1] R. Barry and P. Humblet, "On the number of wavelengths needed in WDM networks," in *LEOS Summer Topical Meeting*, (Santa Barbara, CA), pp. 21–22, LEOS

'92, Aug. 3-5 1992.

[2] R. Barry and P. Humblet, "On the number of wavelengths and switches needed in all optical networks," *To appear in IEEE Trans. on Comm.*, 1993.

[3] R. Ramaswami, "System issues in multi-wavelength optical networks," in *Thirty-first Annual Allerton Conference on Communication, Control, and Computing*, Sept. 1993.

[4] A. Aggarwal, A. Bar-Noy, D. Coppersmith, R. Ramaswami, B. Schieber, and M. Sudan, "Efficient routing and scheduling algorithms for optical networks," tech. rep., IBM Research Report RC 18967 (82799), June 1993.

[5] B. Mukherjee, "WDM-based local lightwave networks part I: Single-hop systems," *IEEE Networks*, pp. 12–27, May 1992.

[6] Y. Birk, N. Linial, and R. Meshulam, "On the uniform-traffic capacity of single-hop interconnections employing shared directional multichannels," *IEEE Trans. on Information Theory*, vol. 39, Jan. 1993.

[7] Y. Birk, F. Tobagi, and M. Marhic, "Bus-oriented interconnection topologies for single-hop communication among multi-transceiver stations," in *Proc. IEEE INFOCOM '88*, pp. 558–567, Mar. 1988.

[8] G. Sasaki and G. Pieris, "A linear lightwave Beneš network," *to appear in IEEE/ACM Transactions on Networks*, 1993.

[9] J. Bannister, M. Gerla, and M. Kovačević, "An all-optical multifiber tree network," *Journal of Lightwave Technology*, vol. 11, pp. 997–1008, May/June 1993. Special Issue on Broadband Optical Networks.

[10] C. E. Shannon, "Memory requirements in a telephone exchange," *Bell System Technical Journal*, pp. 343–349, 1949.

[11] V. Beneš, *Mathematical Theory of Connecting Networks and Telephone Traffic.* Academic Press, New York, 1965.

[12] R. Barry and P. Humblet, "The throughput of wavelength routing networks," in *IEEE LEOS 1993 Annual Meeting*, (San Jose, CA), LEOS '93, Nov. 15-19 1993.

[13] R. Barry, "An all-optical non-blocking $M \times M$ switchless connector with $O(\sqrt{M \log M})$ wavelengths and without wavelength changers," *IEE Electronics Letters*, vol. 29, pp. 1252–1254, July 1993.

[14] R. Barry, *Wavelength Routing for All-Optical Networks.* PhD thesis, Department of Electrical Engineering and Computer Science, MIT, 1993.

[15] P. Feldman, J. Friedman, and N. Pippenger, "Wide-sense nonblocking networks," *SIAM Journal of Discrete Mathematics*, vol. 1, pp. 158–173, May 1988.

[16] S. Alexander *et al.*, "A precompetitive consortium on wideband all-optical networks," *Journal of Lightwave Technology*, to appear in special issue on broadband optical networks 1993.

[17] C. Brackett, "Dense wavelength division multiplexing networks: Principles and applications," *Journal on Selected Areas in Communications*, vol. 8, pp. 948–964, Aug 1990.

[18] R. Barry and P. Humblet, "Latin routers, design and implementation," *Journal of Lightwave Technology*, vol. 11, May/June 1993. Special Issue on Broadband Optical Networks.

[19] R. G. Gallager, "Fixed composition arguments and lower bounds on error probability." Supplementary Lecture Notes for Course 6.441, Massachusetts Institute of Technology, Apr. 1992.

[20] R. Pankaj, *Architectures for Linear Lightwave Networks.* PhD thesis, Department of Electrical Engineering and Computer Science, MIT, 1992.

Comparison of Feed-Forward and Feedback Configurations of the *Staggering Switch* Architecture

Zygmunt Haas
AT&T Bell Labs, Room 4F-501, P. O. Box 3030, Holmdel, NJ 07733

ABSTRACT *Staggering Switch* is a packet contention resolution scheme based on a set of different-in-length delay lines. The major motivation for the *Staggering Switch* is the lack of large, fast, and cost-effective optical random access storage. The *Staggering Switch* contention resolution mechanism comes to avoid the use of the buffers in the traditional meaning of the word. There are several possible configurations of the *Staggering Switch* architecture. In this paper, we compare two such configurations: the *feed-forward* and the *feedback* configurations. In particular, we investigate the probability of packet loss, based on practical considerations such as the size of the optical switching fabrics. Some additional properties of the *Staggering Switch* are also discussed, making the architecture so much more attractive for wide-area all-optical connectivity.

1. INTRODUCTION

Several approaches were proposed to circumvent the difficulty associated with the unavailability of fast and large all-optical random-access memories for implementation of all-optical packet switches. One such approach, termed the *Staggering Switch* ([1, 2]) relies on a set of different-in-size optical delay lines (implemented with optical fibers). Packets arriving to the switch that are destined to the same output (i.e., contending packets), are delayed by a different amount of slots (in the case of synchronously operating switch) or by a different time period (in the case of an asynchronous operation), thus "staggering in time" the contending packets.

The *Staggering Switch* is a general scheme that can be used in different packet switch designs. In particular, the packet switch designs may or may not include feedback in their design. More precisely, and as defined here, in the *feedback* configuration, an optical signal or its replica may transverse an optical device more than once. In the *feed-forward* configuration, no device is transversed twice by the same signal. In this paper, we compare the two architectures.

This work was stimulated by several papers in the technical literature claiming to address the above comparison. However, erroneously, these papers in their comparison rely on metrics borrowed from the "electronic" domain. Because the optical and the electronic domains differ so much, the conclusions presented in these papers are meaningless, at best. In this paper, we intend to dispel some of the myths regarding the comparison of the feed-forward and the feedback-based *Staggering Switch* configurations and show that, under some set of assumptions, the feed-forward configuration can be superior to the feedback-based configuration for implementation of optical switches. In particular, we compare the

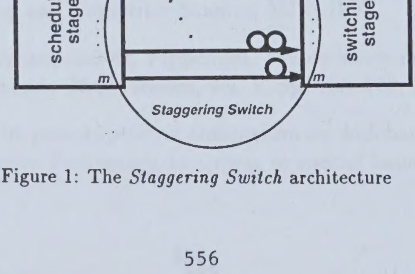


Figure 1: The *Staggering Switch* architecture

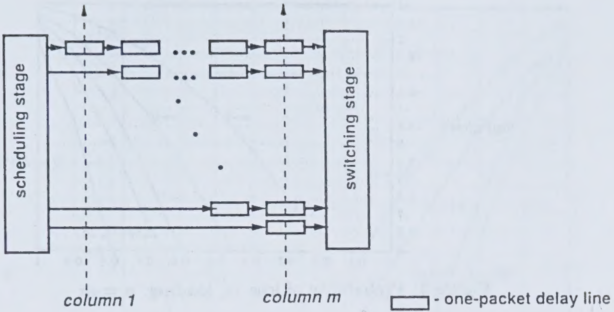


Figure 2: Definition of routing algorithm "data structure"

complexity of implementation of the two configurations, given a constant level of performance (i.e., packet-loss probability). In general, the feedback configuration with no constrains experiences better packet loss performance than the feed-forward configuration. However, with practical constrains (such as limiting the number of recirculations in the feedback elements), the picture is totally changed, showing advantage to the feed-forward configuration in most cases.

Additionally, in this paper we notice with interest the applicability of the *Staggering Switch* architecture to electronic packet-switch design.

2. THE *STAGGERING SWITCH* ARCHITECTURES

2.1 THE *STAGGERING SWITCH* OPERATION: AN OVERVIEW

The *Staggering Switch* architecture uses a set of delay lines, with different delays in general, for the purpose of collision resolution. The collision resolution is performed by delaying the packets with the same destination number by a different number of slots, in the case of synchronously operated switch, or by a different delay period (in the case of asynchronous operation). Thus packets emerging from the collision resolution delay lines have less internal destination collisions.

The collision-resolution delay lines usually interconnects two optical switching fabrics, as shown in Figure 1. We refer to the two fabrics as two stages: the *scheduling stage* and the *switching stage*.[1] Each one of the stages is a reconfigurably nonblocking switching fabric. These stages are implemented with electronically controlled optical devices (e.g., $LiNbO_3$ ([3]). The scheduling stage is connected to the switching stage by m delay lines, d_i, $i = 1$ to m. The delay of the d_i delay line equals i packets.

The energy of each one of the n input lines to the switch is split immediately after its arrival; a "small" fraction of the energy is passed to the *detector*, converted to an electrical signal, and forwarded to the *Control*. The Control reads the header bits to determine the required routing for the packet, and drives the scheduling and the switching stages of the switch.[2]

Additional delay between the splitter and the scheduling module may be needed to compensate for the electronic Control processing time. It is assumed that all the packet are fixed size (i.e., equal time duration), τ [sec]; that time is slotted (i.e., each slot duration is τ); and that arrival of all the packets to the switch is synchronized.

Both the stages, the scheduling stage and the switching stage, may be implemented as *rearrangeably non-blocking* networks; for example Beneš networks ([4]) with dilation ([5]).

The purpose of the scheduling stage is to distribute the packets arriving on the switch input to the delay lines, in such a way that, at any time slot, no two packets arriving at the switching stage are destinated to the same output. In other words, the output collisions present in the inputs are resolved by delaying the colliding packets by different number of slots, so that when they arrive to the switching module, the collisions are resolved.

The scheduling is done by the Control. The Control receives the header information from all the arriving packets and attempts to allocate as many of them as possible into the delay lines. The scheduling is done according to a scheduling algorithm. To perform this algorithm, the Control needs to maintain

---

[1] Note, however, that this is a conceptual description only. In some cases the two stages are actually implemented as one physical fabric, as in the case in Figure 8.

[2] For simplicity, in this paper we've omitted the control connections in all the Figures.

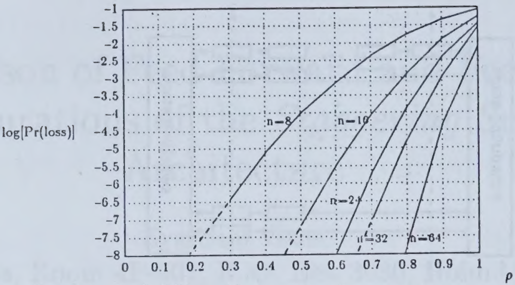


Figure 3: Probability of loss vs. loading: $n = m$

knowledge of the "content" of the delay line at any time. Based on this information, the Control can reduce the number of collisions at the switching stage. Because of the statistical properties of the arrival process, some packets cannot be accommodated (without violating the "no collisions at the switching stage" principle) by the scheduling process. Thus there is some probability of "output collision." This is referred to here as "probability of loss." The purpose of the switching stage is to permute the outputs of the *delay lines*, so that the packets emerge at the required switch output.

For the purpose of the explanation of the operation of the *Staggering Switch*, we define a data representation of the collision resolution delay lines in Figure 2. Each delay line is pictured as a sequence of slots, each slot containing a single packet. The slots are arranged in columns, as shown in Figure 2. The distribution of packets arriving to the collision resolution delay lines is performed by a *scheduling algorithm*. We consider here the following example of the scheduling algorithm:

```
while () do
    for input=1 to $n$ do
    if ($input$ is busy) then begin
        $line := 1$;
        while ($line <= m$) do
                if ($delay - line_{line}$ is busy) then $line := line + 1$
                else if (no other packet to the same $destination$ in the $line$ column)
                    then insert $input$ packet on $delay - line$, break;
        end;
```

In other words, the algorithm scans the inputs sequentially, for each input packet the algorithm tries to insert the packet in the lowest possible $delay - line$, subject to two conditions:

1. that no previous packet was inserted in the $delay - line$ in this time slot; i.e., the output is free,
2. that no other packet to the same *destination* exists in the column that the packet is to be inserted in.

This algorithm is referred to *sequential biased assignment.* Other scheduling algorithms and variation of the presented algorithm are possible. The comparison of performance of the different algorithms is subject to current study.

In evaluating the loss performance of the switch, the following were assumed:

1. The destination addresses are uniformly distributed.
2. The traffic is not time-correlated; i.e., *non-bursty* traffic.
3. There is no correlation between packets arriving on any two inputs.
4. *Sequential biased assignment* is performed.
5. All packets have the same routing- and loss-priorities.[3]

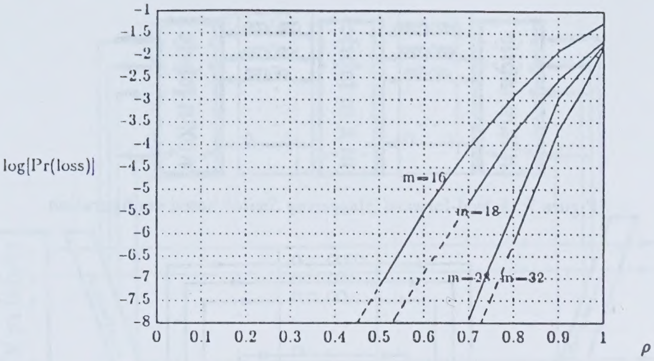


Figure 4: Probability of loss vs. loading: $n = 16$

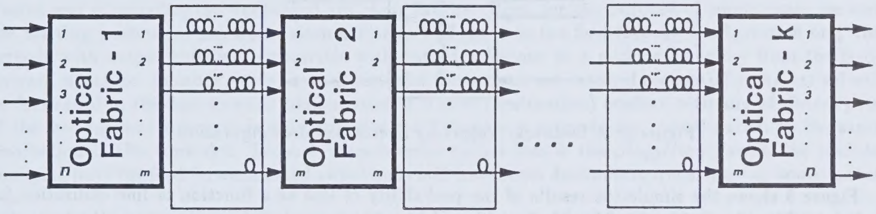

Optical Fabric - 1
Optical Fabric - 2
Optical Fabric - k
1
2
3
n
m
. . . .

Figure 5: An example of the *Staggering Switch* architecture: A feed-forward configuration

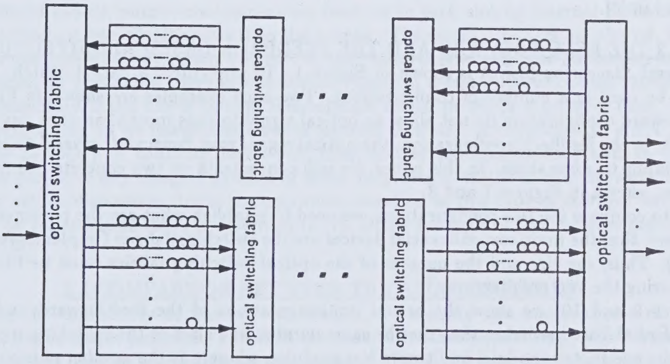

optical switching fabric
optical switching fabric
optical switching fabric
optical switching fabric
optical switching fabric
optical switching fabric

Figure 6: An example of the *Staggering Switch* architecture: A feedback configuration

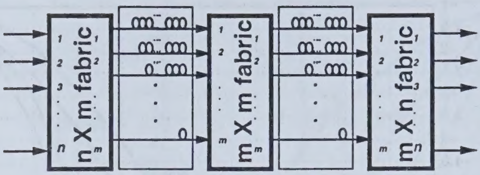


Figure 7: A feed-forward *Staggering Switch*-based configuration

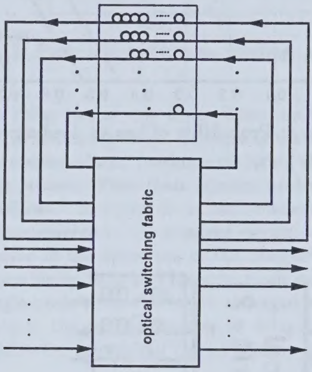


Figure 8: A feedback *Staggering Switch*-based configuration

Figure 3 shows the simulation results of the probability of loss as a function of line utilization for switches of size 8 x 8, 16 x 16, 24 x 24, 32 x 32, and 64 x 64. As evident from the graph, the larger the switch, the lower is the loss probability for a given line utilization. Furthermore, increasing the number of delay lines lowers the loss probability. This is shown in Figure 4, where n is kept constant at 16 and m is varied from 16 to 32. In general, an increase in ρ, in n, and in m results in an increase in the average packet delay through the switch. More information on performance evaluation of the *Staggering Switch* can be found in [6].

2.2 THE FEED-FORWARD AND THE FEEDBACK-BASED ARCHITECTURES

The general *Staggering Switch* is shown in Figure 1. To implement a packet switch, the *Staggering Switch* can be used in a number of configurations. Two such examples are shown in Figures 5 and 6. The feed-forward configuration do not allow an optical signal to pass more than once any element of the switch, while in the feedback configuration, the optical signal may *recirculate* several times by being fed back as an input to some stage. In this paper, we will concentrate on two concrete examples of the two architectures, shown in Figures 7 and 8.

In order to compare the two configurations, we need to establish what are the parameters or interest. We claim here, that the main cost-influential devices are the switching fabrics (implemented, for example, in $LiNbO_3$). Thus, the size and the number of the optical switching fabrics must be maintained equal when comparing the two architectures.[4]

In Figures 9 and 10, we show the actual implementations of the feed-forward and the feedback *Staggering Switch*-based switches that use the same number and sizes of the switching modules; i.e., one $n \times n$ module, one $m \times m$ module, and two $m \times n$ modules, where n is the number of inputs and outputs of the optical switch, and m is a parameter (indicating the size of the *Staggering Switch*). The "1-to-2 selector" and "2-to-1 multiplexor" are simple one-by-two and two-by-one switches, respectively. "m 1-to-2 selectors" are m of such switches operating independently. (The control connections are omitted from

---

[3] To be discussed latter.

[4] In some papers, the number and the size of the delay lines is taken as the main parameter. We claim here that the number of delay lines per se is not of any interest; so is not the length of the delay lines, since the optical fiber consists of a negligibly small fraction of the cost of the switching modules.

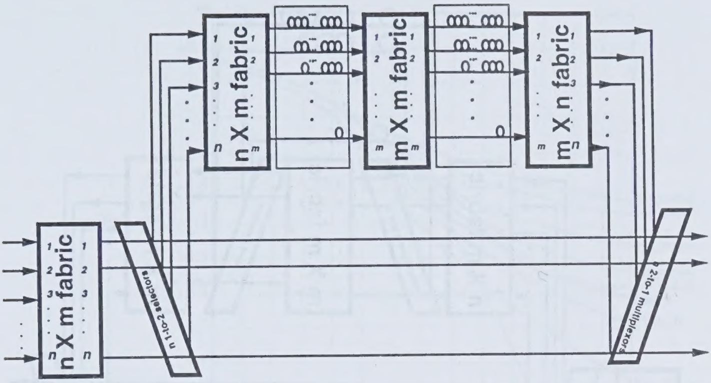


Figure 9: Actual implementation of the feed-forward configuration

the Figures.)

The feed-forward based *Staggering Switch* configuration can be built by cascading the basic *Staggering Switch* and eliminating the duplicated switching fabrics. Here, for the purpose of comparison, we sort the arriving packets for any destination collisions and route to the feed-forward configuration only the arrivals with output collisions or arrivals with output collisions to a packets emerging from the feed-forward structure. In other words, in each time slot the arrivals are scanned sequentially. An arrival will be forwarded to the feed-forward configuration if it is in (destination) conflict with any of the outputs of the feed-forward structure in this time slot, or if there was a previously routed packet to the same destination in this time slot. Because there is some packet loss in the *Staggering Switch*, the packets emerging from the feed-forward based switch may still have some destination collisions; i.e., some packet loss.

In the feedback-based *Staggering Switch* conceptually operates in the same way as the feed-forward example, except that the packets that are in destination collision are fed back to the *Staggering Switch*, and not removed from the switch as in the previous case.

It is interesting to notice that with all the delay lines equal to one slot, the feedback-based architecture is, in principle, similar to the electronic switching architecture, commonly referred to as *shared buffers*. In a shared buffered switch, output contentions are resolved in each slot by internal buffering of all the colliding packets, except one which makes it to the output. The internal buffers are shared, in the sense that each buffer can potentially store any packet[5]. In the "electronic world," the buffers are implemented with RAM devices. In photonics, there is still limited choice of a fast massive optical random-accessed devices. One possibility is the use of optical recirculation loops, in which a piece of fiber, typically of the length corresponding to an integral number of packets or slots,[6] is connected to a 2 × 2 switching module (such as $LiNbO_3$). A colliding packet can, thus, be inserted into the loop and, in principle, be "stored" for the period of any integral number of slots. In practice, the fiber and the switching module attenuation, as well as dispersion, limits the maximum duration of this type of storage. An example of a switch based on recirculation loops is described in [7]. Other example of the feed-forward architecture is presented in [8].

2.3 COMPARISON BETWEEN THE ARCHITECTURES

In comparing the complexity of the two approaches, we identify the maximal size of the optical switching fabric as the major criterion. This is in contrast with other papers (e.g., [9]), where the length of the delay lines was used as a criterion for comparison.[7] We claim here that the length of the delay lines, being implemented with optical fiber, only marginally affects the total cost of the switch design, and is, therefore, a meaningless practical comparison criterion.

To compare the two configurations, we've computer-simulated the two configurations and measured the probability of loss in the two cases. The simulations were done at utilization 0.9 (i.e., $\rho = 0.9$;

---

[5] From/to any source/destination

[6] In particular, designs having loops of length corresponding to a single packet have been proposed

[7] Actually, the total storage capacity in the optical delay lines was used for comparison in these papers.

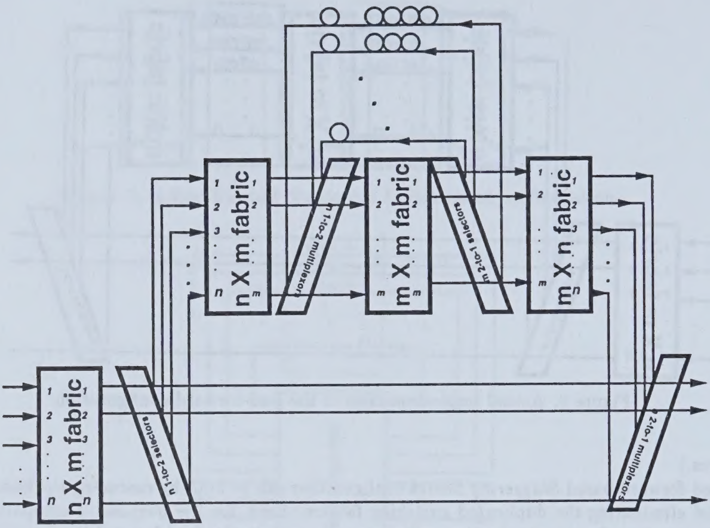


Figure 10: Actual implementation of the feedback configuration

utilization being the probability that a given slot is occupied) and for two switch sizes: an 8 × 8 switch ($n = 8$, Figure 11) and a 16 × 16 switch ($n = 16$, Figure 12). In the above Figures, the curves "2-stg_ff" and the "recirc" correspond to the configurations in Figures 9 and 10, respectively. As displayed in these performance graphs, the feed-forward configuration outperform the feedback configurations for small collision-resolution module sizes (m). As m increases, the feedback configuration becomes considerably better. (The curve "2.5-stg_ff" in Figure 11 corresponds to performance of the feed-forward configuration when the packets are allowed to be removed after the first *Staggering Switch* and forwarded directly to the output through a set of "1-to-2 selectors" and "2-to-1 multiplexors.")

In evaluating the performance of the feedback configuration, it was assumed that there is no limit on the number of time an optical signal may be fed back into the *Staggering Switch*. In practical designs, this is not the case, since after some number of recirculations the degradation of the optical signal (signal attenuation, spectrum dispersion, and cross-talk accumulation) renders the signal unusable. Thus, the number of recirculations needs to be limited.

To demonstrate the effect of limiting the number of recirculation in the feedback configuration on the packet loss probability, we've further simulated this configuration restricting the number of recirculations (termed here "life_time") to 1, 2, 4, 6, and 8. The corresponding results are shown in Figures 12 and 11. As can be seen from these graphs, the feedback configuration experiences better performance then the feed-forward case when the maximum number of recirculations is at least 8 or 6, in the $n = 8$ and $n = 16$ cases, respectively. Simple calculations of attenuation show that maximum number of recirculations need to be limited to 1-2, before optical amplification is required. Moreover, the cross-talk accumulation may limit the number of recirculations to about 4. Thus, we conclude that in a practical design, the feed-forward configuration outperforms the feedback configuration.

Additionally, we notice that the feedback configuration suffers from the following drawbacks:

- There is the need for a mechanism to remove the "dead" packets that recirculate longer than the maximum allowable recirculations.
- The packets emerging from the switch may have relatively large dynamic range.
- There is the need to count and keep track of the "life time" of each packet.

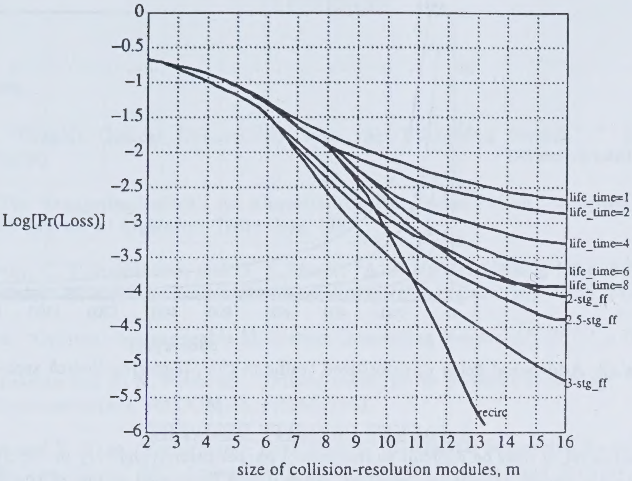


Figure 11: Performance comparison for $n = 8$

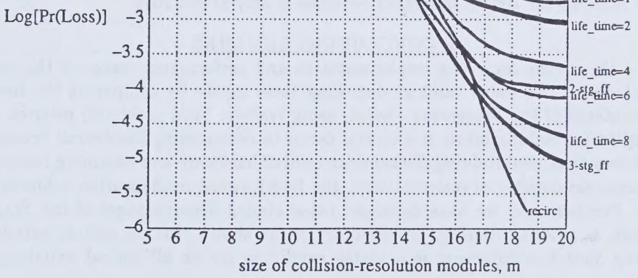


Figure 12: Performance comparison for $n = 16$

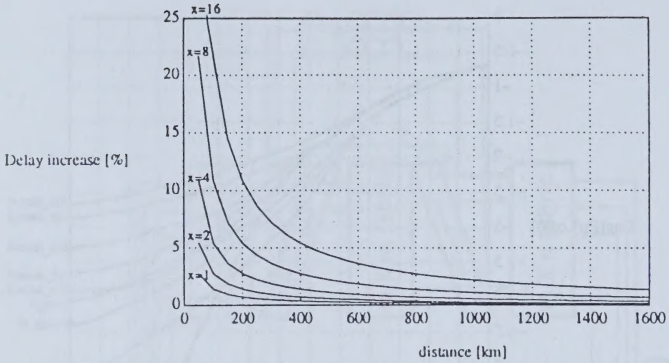


Figure 13: Additional delay of prioritized traffic in the *Staggering Switch* architecture

3. PACKET PRIORITY DELIVERY

As mentioned in [9], it may be difficult to implement packet priority delivery in the *Staggering Switch* architecture, since a packet cannot be "recalled" after it was forwarded to one of the delay lines. This shortcoming is present in both, the feed-forward and the feedback configurations (but is not present in the recirculation loop architecture, since packets can be kept longer in a loop to allow a higher priority packet to preempt them).

This lack of packet priority delivery results in increased delay for a higher priority packets. Here, we would like to evaluate this penalty. In particular, we are exploring this penalty for the ATM-based switching system. Furthermore, we assume a wide-area networking, where the distance between switching nodes is on the order of hundreds of kilometers. Figure 13 depicts the increased delay (in percentage) as a function of distance between the switches. The Figure is calculated for the 16×16 switch and the parameter x is the expansion ratio of the number of the delay lines relative to the switch size; i.e., $m = n \cdot x$. The essence of the phenomenon is that as the distance between switches increases, the propagation time of the signal consists of a bigger portion of the total end-to-end delay (the total end-to-end delay is the cell transmission time, the switching delay, and the inter-switch propagation time), reducing the effect of the switching delay. Consequently, as the distance increases, the penalty in delay due to the inability of the feed-forward architecture to move the higher priority packets ahead of the lower priority packets is relatively small. For example, for the 16×16 switch with $m = 128$, the additional delay increase for 100km separated switches is only about 20%.

6. CONCLUDING REMARKS

Stimulated by the myths regarding implementation and performance issues of the optical *Staggering Switch* architecture, this paper aims at dispelling these myths by comparing the feed-forward and feedback configurations of the *Staggering Switch*, using realistic (and practical) metrics. In particular, the size of the optical switching fabric is a crucial factor in determining the overall "complexity" of the design. It was shown that, maintaining the same amount of hardware and assuming reasonable practical limits to the maximum number of recirculations, the feed-forward configuration achieves better packet loss probability. Furthermore, we have discussed other eluded disadvantages of the *Staggering Switch* architectures, such as priority routing. In summary, we've shown that an optical switch design based on the *Staggering Switch* architecture is a viable candidate for an all-optical switching in wide-area networks.

The advocated here *Staggering Switch* architecture can be also used for non-optical implementations. One can envision a VLSI design of a fast and wafer space-efficient shift register with no access to intermediate stages. Such shift registers may be used for the delay-lines implementation in the *Staggering Switch* architecture. The consideration is whether the wafer space savings in the designs of the "shift registers" allows larger expansion ratio (i.e., $\frac{m}{n}$) and thus better performance (i.e., reduced packet loss probability) as compared with the implementation of random access memory. Alternatively, keeping the packet loss probability constant, the reduction in the wafer space for the *Staggering Switch* architecture may allow building larger switch size, as opposed to other switching architectures.

7. REFERENCES

References

[1] Z. Haas, "Gigabit Optical Connectivity with the 'Staggering Switch,' " paper 1787-16, OE/FIBERS'92.

[2] Z. Haas, "The 'Staggering Switch': An Almost-all Optical Packet Switch," accepted for publication in the *IEEE Journal of Lightwave Technology,* 1993.

[3] T. O. Murthy, C. T. Kemmerer, and D. T. Moser, "A 16 × 16 Ti:$LiNbO_3$ Dilated Benes Photonic Switch Module," *Photonic Switching,* Postdeadline paper, Salt Lake City, Utah, March 6-8, 1991.

[4] V. E. Beneš, "Optimal Rearrangeable Multistage Connecting Networks," *BSTJ,* p.1641, July 1964.

[5] K. Padmanabhan and A. N. Netravali, "Dilated networks for photonic switching," *IEEE Transactions on Communications,* vol.COM-35, p.1357, 1987.

[6] R. Izmailov and Z. Haas, "Performance Evaluation of the Optical Staggering Switch in Underload Input Traffic Conditions," 26^{th} *Annual Conf. on Information Sciences and Systems,* Princeton, NJ, March 1992.

[7] J. Spring and R. S. Tucker, "Photonic 2 x 2 Packet Switch with Input Buffers," *ELECTRONICS LETTERS,* 4^{th} February 1993, vol. 29, no. 3.

[8] D. K. Hunter and I. Andonovic, "Optical Contention Resolution and Buffering Module for ATM Networks," *ELECTRONICS LETTERS,* 4^{th} February 1993, vol. 29, no. 3.

[9] M. J. Karol, "Queueing in optical packet switches," *SPIE Multigigabit Fiber Communications 1992,* Vol. 1787, Boston, MA, September 8-9, 1992.

Space-Time Networks for Optical Communications

Aruna V. Ramanan, Harry F. Jordan and Jon R. Sauer
Optoelectronic Computing Systems Center
Department of Electrical and Computer Engineering
University of Colorado at Boulder, Boulder, CO-80309

Abstract

Fiber optic networks lack an efficient mechanism for storage at intermediate nodes. Hence, fast packet-switching in such networks requires the packets be continuously moved until they reach their destination. The deflection routing strategy eliminates the need for static buffering by allowing one of the contending packets to exit through the desired port, and deflecting the others out through available ports. The deflected packets are forced to take a longer route to their destinations. We have found that by simultaneously switching in both space and time, using optical multi-spatial channel, multi-time channel time slot interchangers, the contention for spatial channels can be greatly reduced. Limited temporal reordering at switch nodes is sufficient to enhance the performance of an uniformly loaded network to near ideal behavior for the topology. In this paper we show that a space-time network has a greater capability to support bursty traffic, a typical characteristic of communication systems, than a purely spatial network.

1 Introduction

Fiber optic communication has been used successfully for point to point communication in telecommunications and local data links. Until recently, the inherent advantages of optoelectronics for intelligent networks – more than simple transmission – has been underexploited. The basic requirement placed on the network is to provide an efficient connection that satisfies the communication requirements of the electronic hosts. The functions to be carried out by an optical communication network are the electro-optic conversions at the sources and receivers, transmission, and switching. Since, purely optical switching is currently not possible, photonic switching utilizes the combined strengths of optics and electronics, through electronically controlled switching of optical signals. The network may be circuit-switched or packet-switched according to the communication requirements.

Both circuit and packet-switched networks can be implemented using wavelength-division multiplexing (WDM) schemes that allow different transmissions at different wavelengths to be multiplexed concurrently on the same fiber [2]. Packet switching in fiber optic networks can be performed in a variety of ways. While electronic switching systems can switch in space and time domains, optical systems can switch in the wavelength-domain too, thus providing a greater range of switching techniques [3]. This added flexibility of optical systems is currently being exploited to build new fiber optic packet-switched systems. The low loss bandwidth of fiber is about 30 THz [1], with perhaps only 100 GHz practically accessible. Guided wave transmission through optical fibers is suitable for larger distances due to the large distance-bandwidth product of optical fibers.

The high bandwidth of fibers can be exploited to provide multi gigabit/s communication by transmitting bits that belong to short packets in parallel using bit per wavelength encoding [4]. A fiber optic packet-switched system for multicomputer communications based on this paradigm is presented in [5]. The switch nodes are flow-through 2×2 switching structures that separate the payload data from control data and forward the control data to an electronic pipelined control processor that sets the control for the 2×2 switch[6]. The payload data is maintained in optical format and delayed in a fiber delay line to compensate for control processor latency. The switch nodes contain no static buffers. In case of contention, one of the two contending packets will win the contention and the other will be sent out through the other port. This routing strategy, called deflection/hot-potato routing [7, 8] is particularly suitable for fiber optic packet-switched networks, since it is currently not possible to provide efficient optical buffers for storing and forwarding packets that lose contention. Conversion of optical data into electronics at nodes for storage and reconversion to optics for transmission will necessarily slow the network. A similar bufferless network that implements deflection routing, and utilizes a single wavelength to transmit bits that belong to a packet bit-serially at very high data rates, is presented in [9].

Deflections occur due to contention for spatial channels and cause deflected packets to take a longer path to their destinations. Schemes have been proposed [10] to minimize the negative effects of deflection routing using some architectural changes to the network. Our approach is to reorder packets in time, dynamically during flight, so as to reduce the contention for spatial channels. This paper shows that by switching in the time domain in addition to the spatial domain using multi-spatial channel time-slot interchangers, the performance of deflection routing networks can be greatly improved. We restrict our attention to 2 Space channel, 2 Time channel (2S2T) switch nodes in order to reduce the complexity of the node and maintain fast packet-switching. Implemented with appropriate and foreseeable photonic devices, the building block switching nodes described here should outperform any all-electronic system in several respects.

2 Purely Spatial Deflection Routing

Deflection routing requires a topology in which there is at least one path from every node to every other node, so that deflected packets can eventually reach their respective destinations. A variety of topologies such as Shuffle Exchange network(SX Net), ShuffleNet, Toroidal Net, and Hypercubes satisfy these requirements. The effect of deflections is an increase in source to destination transit time through increased residence time in the network. The penalty for each deflection is dependent on the network topology. The existence of multiple minimum distance paths between some source-destination pairs in networks such as the Toroidal Net and the ShuffleNet introduces the *don't care* condition for routing at the nodes. In such networks, every node has a set of destination nodes that can be reached in minimum time via either port. If α is the link utilization, and β the probability that an incoming packet will care about the output port in order to reach its destination in minimum time, then the probability of deflection in a uniformly loaded network under random contention resolution is half the probability of facing a contention and is given by

$$p_d = \frac{p_{cont}}{2} = \frac{\alpha\beta}{4} \tag{1}$$

β will depend on the load as well as the topology of the network.

The maximum probability of deflection is 1/4 and occurs in fully loaded ($\alpha = 1$) single minimum distance path networks ($\beta = 1$). If more minimum distance paths are available the factor β in Eqn. (1) will go down, bringing down the probability of deflection. Other considerations, such as minimizing the average internode distance and reducing the penalty for deflection, make it difficult to choose a topology in which the probability of deflection is very small. So, we cannot bring down the probability of deflection arbitrarily by choosing or designing a topology without compromising on other factors.

3 Deflection Routing under Spce-Time Switching

Single [11] and multiple [12, 13] spatial channel time-slot interchangers perform arbitrary permutation of time-slots within nonoverlapping time frames. We consider a sliding time window in our 2S2T switch, so that a contention for output port at a switch node can be resolved either by the preceding pair of packet slots or by the following pair of packet slots, if they have the capability to do so. Hence a sliding time window provides greater contention resolution capability to the switch node than fixed time frames.

One possible configuration for a 2S2T deflection routing node is shown in Fig. (2). The 2×2 deflection routing node is the conventional spatial node with additional capability to compute the control settings for the space-time permuter. In this configuration, the 2×2 deflection routing node will perform contention resolution between packets arriving simultaneously, enforcing the priority scheme built into it. In case of a deflection which is not repaired by the space-time permuter, the higher priority packet will still remain undeflected. The space-time permuter needs to perform only 3 permutations out of the 24 permutations possible in an universal 2S2T time-slot interchanger. The first permutation is an identity permutation, while the other two require swapping of leading packet slot in one of the two spatial channels with the trailing packet slot in the other spatial channel. It is possible to merge the space-time permuter into the switching stage of the 2×2 deflection routing node. We are currently analyzing the optimal configuration, which will be reported in a future paper on the design of the system.

The temporal reordering of packets in a space-time switch reduces the probability of deflection at each switch node in the network. In a purely spatial network, the probability of deflection in a uniformly loaded network under random contention resolution is half the probability of contention. In a space-time switch, the contention can be removed either by the slot combination on the two spatial channels in the preceding time slot or by the slot combination on the spatial channels in the following time slot. Hence, the probability that a packet will ultimately get deflected is given by

$$p_d = \frac{p_{cont} p_r p_r^{'}}{2} \tag{2}$$

where, p_r and $p_r^{'}$ are the probabilities that the following and preceding pair of slots cannot resolve the contention for the output port faced by the packet. Probability $p_r^{'}$ is slightly more than the probability p_r because the the packet slot combination preceding a new slot combination entering the space-time permuter might have lost its contention resolution power by resolving a contention in the packet slot combination preceding it. The probability of deflection p_d is found to be

$$\begin{aligned} p_d &= \frac{1}{2}.\frac{\alpha\beta}{2}.\frac{\alpha\beta(1-\alpha\beta/4)}{1-(\alpha^2\beta^2/4)(1-\alpha\beta/2)^2}.\alpha\beta(1-\alpha\beta/4), \\ &= \frac{\alpha^3\beta^3}{4}.\frac{(1-\alpha\beta/4)^2}{1-(\alpha^2\beta^2/4)(1-\alpha\beta/2)^2} \end{aligned} \tag{3}$$

A detailed analysis and derivation of the probability of deflection in a 2S2T switch can be found in [14].

Eqn. (3) shows that at full link utilization ($\alpha = 1$), the probability of deflection in a single minimum distance path network ($\beta = 1$) falls from 0.25 in a 2×2 switch to 0.15 in a 2S2T switch. The decrease in probability of deflection is much more in multiple minimum distance path networks. For example, consider a network running under a uniform load leading to a link utilization of 0.85 and a care probability of 0.8. Under these conditions, the probability of deflection in a purely spatial network is 0.17 and that in a space-time network is only 0.057.

4 Network under uniform load

In this section we present some results on the performance of a ShuffleNet under deflection routing and space-time switching, obtained through simulation, that validates the results of our analysis of the probability of deflection and predicts the various performance parameters. Though we have chosen to present the results on the ShuffleNet similar results have been obtained for the unidirectional Manhattan Street network (MSNet), a two dimensional torus, under uniform load [15].

A ShuffleNet, Fig. (1), is a logarithmic topology with columns of nodes interconnected by a perfect shuffle interconnection with the last column wrapping around on the first. The system with k columns consists of $N = k*2^k$ switches of in-degree and out-degree two into the network. A host is attached to each switch node by means of two bidirectional ports. Potentially, a host can inject and/or receive two packets into/from the switch node every network cycle. A newly generated packet can enter the network, if either one of the incoming links is free or if one of the incoming packets is for the host. Until then the packet is buffered at the the input to the network. It is assumed that enough resources are available to send a packet that has reached its destination node immediately into the host. We assume that all node-link pairs are of constant length. We express the flight latency of packets in *hops*, i.e. the number of node-link pairs traversed.

In a ShuffleNet many source-destination pairs have multiple minimum distance paths between them. In this network the minimum number of node-link pairs between a source and a destination, d, varies from from 1 to $2k - 1$. A packet will care about the output port assignment only at the last $min(d, k)$ nodes. As a result, a packet that gets deflected will enter a *don't care* state until it is again at a distance of k hops from its destination. Thus, the probability of care β is less than 1. For the network sizes that we simulated β varied between 0.6 and 0.8, causing a significant reduction in the probability of deflection at the space-time node as opposed to the purely spatial node.

Figures (4) and (5) show the average flight latency and the throughput as a function of link utilization, obtained through simulation of a 2048-node ShuffleNet with 2S2T nodes as well as 2×2 nodes. Up to moderately high loads, the wait buffer component of latency is negligible as compared to the flight latency of packets. The flight latency of a 2S2T node remains close to ideal, particularly up to moderate link utilizations. The reduction in flight latency leads to increased throughput as seen in Fig. (5).

Another important improvement in the network behavior due to time-domain switching is in the flight latency distribution of packets. In a deflection routing network, the tail of the delay distribution tends to be long if random contention resolution is used at the nodes. Any priority scheme, such as providing preference for the colliding packet that has been deflected more, helps to shorten the tail of the distribution. Though the tail is shortened, the average behavior of the network does not change significantly. Figures (6) and (7) show the latency distribution histogram for a 2048-node ShuffleNet with 2×2 nodes and 2S2T nodes under the same injection rate and age-priority scheme outlined above. In each case, the ideal distribution, under no deflections, has also been plotted for comparison. Apart from reducing the average latency, a 2S2T node also shortens the tail of the latency distribution, i.e. it reduces the variations among the latency of packets.

5 Performance under bursty traffic

The traffic in communication networks tends to be bursty. There are wide variations in the size and frequency of the bursts. Simulation results indicate that a network under space-time switching adapts more efficiently to temporal bursts in traffic as compared to a similar network under purely spatial switching. The only constraint is that the average load presented to the network be less than the maximum that can be supported by the network without saturating the links.

The results presented here are under the extreme condition that all hosts periodically turn bursty at the same time. No new requests are generated by a host until all packets of its latest burst enter the network. During every simulation, the system is initialized in a steady state under uniform load before turning the system into a burst mode. Apart from measuring the average throughput and latency of packets received between bursts, the relaxation behavior of the system following a burst was also studied. As soon as the hosts turn bursty, they potentially want to inject one packet per tick until all the packets of their respective bursts are inserted into the system. Since, the network cannot accept packets at that rate, the hosts are blocked out. The number of host block-outs in the system during intervals of 10 ticks were measured. Figure (8) shows the typical behavior of the system. These results were obtained for the 384 node ShuffleNet system with space-time nodes. The background host insertion bandwidth, η was 0.087 pkts/node/tick, which is equivalent to 80% link utilization for that uniform load. The burst size B was 32 packets. Immediately following the start of a burst, the number of host block-outs rises sharply. Given enough time between bursts, the system ultimately relaxes to the steady state behavior under the uniform background load. If a new burst is started before the system reaches its background steady state, the system still goes into a repetitive behavior. The only difference caused by the more frequent bursts is that the system does not return to the uniform loading steady state behavior at any time. As the burst interval is decreased, the system ultimately gets overloaded and exhibits non-steady behavior. The background insertion bandwidth η and the burst size B determine the minimum burst interval. As long as the burst interval is equal to or greater than this minimum, the network adapts to the bursts by settling into a steady periodic behavior.

The repetitive behavior suggests that over each burst interval the average host insertion bandwidth remains the same. This was indeed observed during simulations, after the system settled into steady periodic behavior under the bursts. The average flight latency was, however, found to be greater than the average flight latency when the system is under a uniform load equal to the average host insertion bandwidth under the bursts. For the 384 node ShuffleNet systems with burst interval of 500 ticks, the increase was within 10%. The reason for this increase is the increase in the total number of deflections due to higher probability of collision when the system is running at nearly full link utilization due to the bursts.

If the system takes T_b ticks to insert a burst of size B into the network, then the effective or average insertion bandwidth of the hosts is given by

$$\eta_{eff} = \frac{(T - T_b)\eta + B}{T}, \tag{4}$$

where T is the burst interval. Equation (4) can be rewritten to express T_b as a function of η, B, T and η_{eff} as

$$T_b = \frac{B + (\eta - \eta_{eff})T}{\eta}. \tag{5}$$

Figure (9) shows T_b, computed using Eqn. (5) using the effective insertion bandwidth obtained through simulations, as a function of burst size for the 384 node ShuffleNet systems without and with space-time nodes. One set of curves is for the system with uniform loads leading to 50% link utilization and the other for uniform loads that lead to 80% link utilization. The time to inject a burst increases slowly with the burst size and becomes linearly dependent on the burst size for larger bursts. This is evident from Eqn. (5), in which the second term in the numerator tends to $(1 - \eta_{max}/\eta)T$ for larger values of B. η_{max} is the maximum insertion bandwidth that can be supported by the system. The superiority of the space-time system over the equivalent spatial system is evident from the fact the bursts are injected more rapidly by the space-time system, thereby allowing larger bursts than an equivalent spatial system. A similar behavior was also observed in MSNet systems.

6 Summary

A deflection routing protocol is essential to take advantage of the enormous potentials offered by fiber optics for packet-switched communications. Simultaneous space and time switching makes packet-switched networks under deflection routing protocol nearly as efficient, with reasonable network parameters, as an ideal network without contention. Limited temporal reordering greatly reduces the contention for output spatial channels at the switching points. The space-time network adjusts better to bursts by allowing longer bursts within the same burst interval, making it more suitable for networks in which the traffic tends to be bursty.

In conventional packet-switched networks under a deflection routing protocol, packets are stored in flight in the network links rather than being buffered statically at the switching point. The space-time switching nodes cause contending packets that would otherwise encounter deflection to be reordered in time if the contention can thus be removed. Effectively, buffering in the network is reduced. At the outset our routing paradigm may appear to be similar to finite buffering in store-and-forward networks. The most important feature that distinguishes the two is that time is brought into the switching paradigm explicitly in a space-time node. While it is possible to provide dynamic output buffering, as in the scheme used in [16], considerably less hardware is required to implement simultaneous space and time switching. Since the required switching logic is simple and pipelined, the network links can be very fast. A time-slot interchange involves a frame delay as well as additional switching delay. These delays will add a few pipeline stages to each switch node without altering the link bandwidth. We find that the increase in latency due to additional delay at 2S2T switch nodes is more than offset by the decrease in latency due to reduced number of deflections even at short internode distances, while maintaining the advantage of increased throughput.

Acknowledgements

This work is supported in part by the US Army grant number DASG 60-91-C-0143, by the National Science Foundation Engineering Research Center grant number ECD 9015128 and by the Colorado Advanced Technology Institute.

References

[1] C. A. Brackett, "Dense wavelength division multiplexing networks: Principles and applications," *IEEE Journal on Selected Areas in Communications*, vol. 8, pp. 948–964, Aug 1990.

[2] "Special issue on dense wavelength division multiplexing techniques for high capacity and multiaccess communication systems," *IEEE Journal on Selected Areas in Communications*, vol. 8, Aug 1990.

[3] H. S. Hinton, "Photonics in switching," *IEEE LTS: Special Issue on Gigabit Networks*, pp. 26–35, Aug 1992.

[4] J. R. sauer, "Multi-Gbit/s Optical Computer Interconnect," in *Proceedings of the SPIE*, Sep. 1991.

[5] J. R. Sauer, D. J. Blumenthal, and A. V. Ramanan, "Photonic Interconnects for Multicomputer Communications," *IEEE LTS: Special Issue on Gigabit Networks*, pp. 12–19, Aug 1992.

[6] D. J. Blumenthal, K. Y. Chen, J. Ma, R. J. Feuerstein, and J. R. Sauer, "Demonstration of a Deflection Routing 2x2 Photonic Switch for Computer Interconnects," *IEEE Photonics Technology Letters*, vol. 4, no. 4, pp. 169–173, 1992.

[7] P.Baran, "On Distributed Communication Networks," *IEEE Transactions on Communication Systems*, vol. 12, pp. 1–9, 1964.

[8] J. S. Kowalik, ed., *Parallel MIMD Computation: HEP Supercomputer and Its Applications.* Cambridge, MA: MIT Press, 1985.

[9] A. S. Acampora and S. A. Shah, "Multihop lightwave networks: A comparison of store-and-forward and hot-potato routing," *IEEE Transactions on Communications*, vol. 40, pp. 1082–1090, june 1992.

[10] A. Krishna and B. Hajek, "Performance of shuffle-like switching networks with deflection," in *Proceedings of IEEE INFOCOM '90, The Conference on Computer Communications*, IEEE Computer Society Press, 1990, pp. 473–480.

[11] S. V. Ramanan, H. F. Jordan, and J. R. Sauer, "A New Time Domain, Multi-Stage Permutation Algorithm," *IEEE Transactions on Information Theory*, vol. 36, no. 36, pp. 171–173, 1990.

[12] H. F. Jordan, K. Y. Lee, and D. Lee, "Multichannel Time Slot Permuters," in K. Annamalai, ed., (*High-Speed Fiber Networks and Channels II*), Vol. Proc. SPIE 1874 - paper 26, Sept 8-9, 1992.

[13] D. K. Hunter, I. Andonovic, D. G. Smith, , and B. Culshaw, "Architecture for Optical TDM Switching," *Electronics Letters*, vol. 28, pp. 1746–1747, Aug 1992.

[14] A. V. Ramanan, H. F. Jordan, and J. R. Sauer, "Space-Time Switching in Deflection Routing Networks," OCS Technical Report No.93-20, OCS Center, University of Colorado, Boulder, CO 80309-0525, 1993.

[15] A. V. Ramanan, H. F. Jordan, and J. R. Sauer, "Space-Time Switching in Fiber-Optic Packet Switched Networks," in *Proceedings of the SPIE, Conference on Multigigabit Fiber Communication Systems*, Vol. 2024, (San Diego, CA), July 1993.

[16] I. Chlamtac and A. Fumagalli, "QUADRO-Star: High Performance Optical WDM Star Networks," in *IEEE GLOBECOM '91*, Dec 1991, pp. 1224–1229.

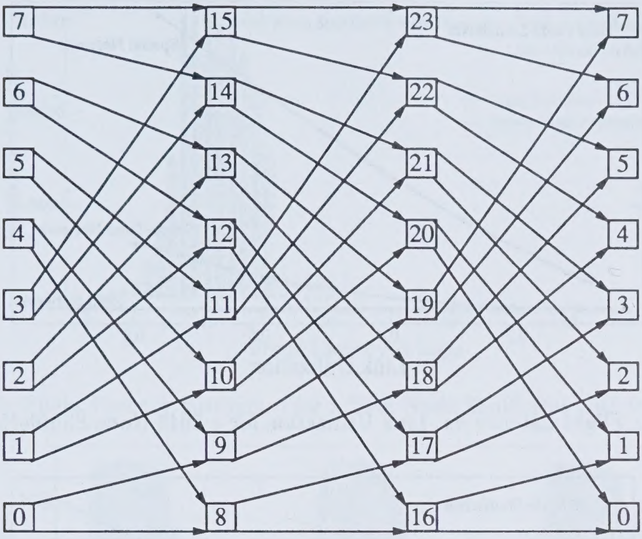

7
6
5
4
3
2
1
0
15
14
13
12
11
10
9
8
23
22
21
20
19
18
17
16
7
6
5
4
3
2
1
0

Figure 1: A 24 node ShuffleNet interconnect

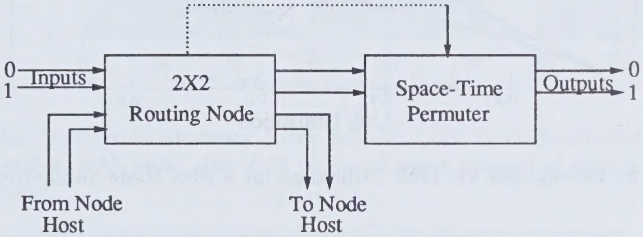

0
1
Inputs
2X2
Routing Node
Space-Time
Permuter
Outputs
0
1
From Node
Host
To Node
Host

Figure 2: A Schematic Diagram of a Space-Time Node

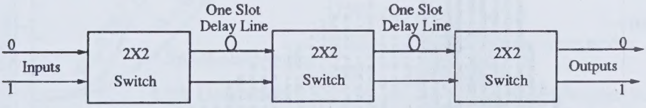

0
Inputs
1
2X2
Switch
One Slot
Delay Line
2X2
Switch
One Slot
Delay Line
2X2
Switch
Outputs
0
1

Figure 3: A 2S2T Space-Time Permuter

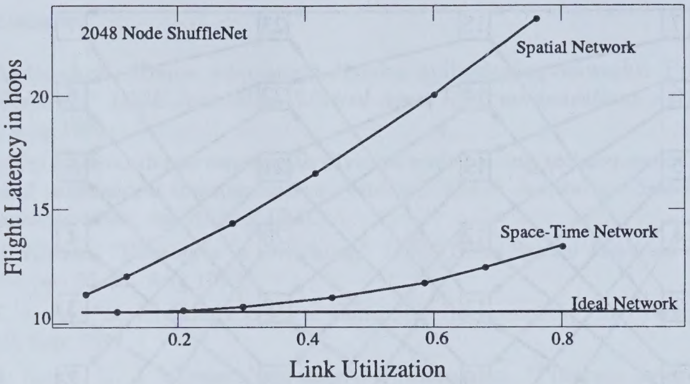


Figure 4: Flight Latency vs. Link Utilization for a 2048 Node ShuffleNet

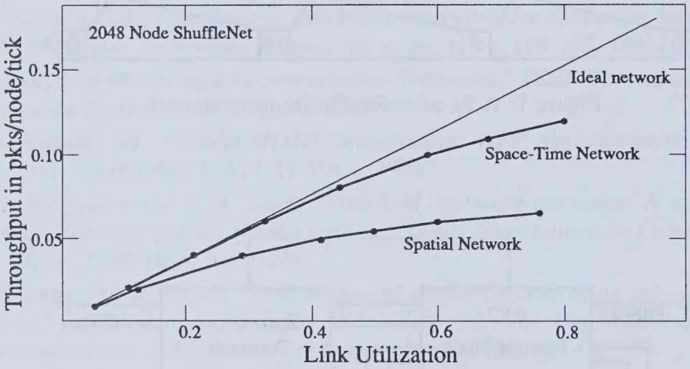


Figure 5: Throughput vs. Link Utilization for a 2048 Node ShuffleNet

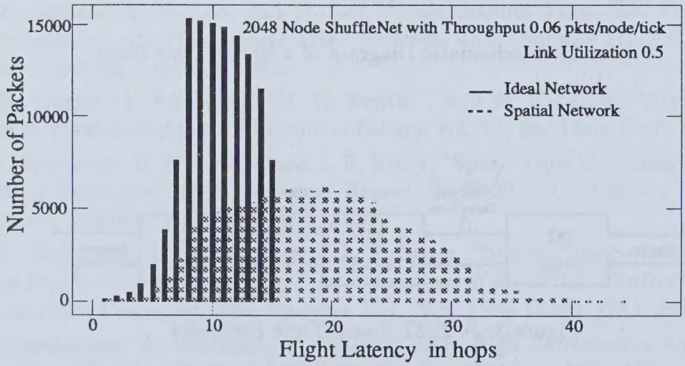


Figure 6: Flight Latency Histogram for a 2048 Node ShuffleNet with 2X2 Nodes

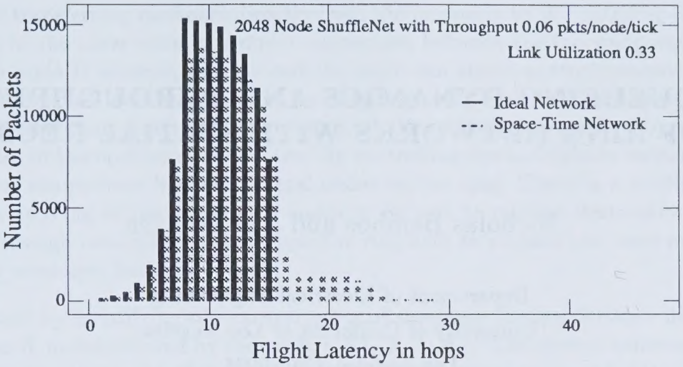


Figure 7: Flight Latency Histogram for a 2048 Node ShuffleNet with 2S2T Nodes

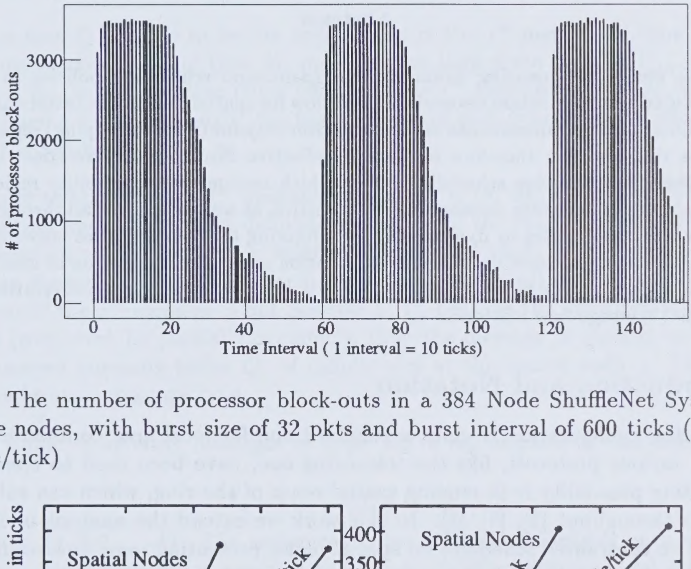


Figure 8: The number of processor block-outs in a 384 Node ShuffleNet System with space-time nodes, with burst size of 32 pkts and burst interval of 600 ticks ($\eta = 0.087$ pkts/node/tick)

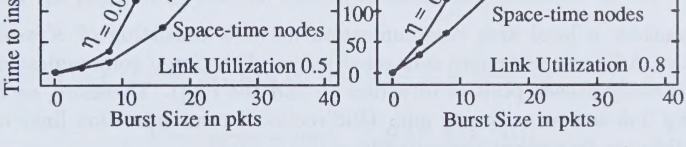


Figure 9: The time to insert a burst into the 384 node ShuffleNet system, with burst interval of 500 ticks for two different link utilizations under uniform background loads

QUEUEING DYNAMICS AND THROUGHPUT OF RING NETWORKS WITH SPATIAL REUSE

Nicholas Bambos and Anh Nguyen

Department of Electrical Engineering
University of California at Los Angeles
Los Angeles, CA 90024

Abstract

We study the queueing dynamics of transmission scheduling policies in ring structured communication networks which allow for spatial reuse. The latter enables the simultaneous transmissions of messages over disjoint (nonoverlapping) segments of the ring and can therefore increase its effective throughput. We construct a *distributed* transmission scheduling policy which maximizes the stability region of the network under given message length statistics, as well as the overall throughput of the ring. It operates by dynamically reconfiguring the connectivities between the nodes on the ring, based on minimal information about the population of messages in the buffers, and is globally fair. The results hold under very general (stationary ergodic) message flow statistics.

1. Introduction and Notation

The ring configuration is quite a standard one for local area communication networks and various protocols, like the token-ring one, have been used to operate them. An interesting possibility is to employ spatial reuse of the ring, which can substantially enhance its throughput [3], [5], [1]. In this work we extend the analysis of Bambos & Nguyen [1] on centralized schedules for spatial reuse, presenting some new results on natural *distributed* message scheduling policies in such networks, which exhibit some optimal properties related to throughput and stability. We largely stick to the notation and problem description of [1]. Regarding the results we only present the key ideas, algorithms and sketches of the proofs, omitting extended work on implementation, simulation and performance evaluation of the algorithms.

We consider a local area communication network consisting of K nodes in ring configuration. Adjacent nodes are connected via *unidirectional* communication links (in a fixed, arbitrarily chosen positive direction around the ring). Therefore, each node has an *incoming* link and an *outgoing* one. (Bidirectional communication links can also be handled within the framework presented here.)

Each node is essentially a special purpose switch that can be in one of two states (modes), *open* or *closed*, at any given time. When being in the *closed* state, it merely acts as a bridge, transferring messages directly from the incoming to the outgoing link. When the node is in the *open* state, the direct connection between the incoming and outgoing links of the node is severed, but instead the node can simultaneously receive messages on its incoming link and transmit on its outgoing one. Existing hardware can easily perform this switching function. The control of the network is exercised by setting the various nodes in the open or closed states. By controlling the node states we can establish simultaneous connections between several nodes on the ring. There is a continuous flow of messages arriving at random source nodes to be sent to random destination nodes on the ring. Through controlled spatial reuse the ring acts as a space and time multiplexer, transferring messages between nodes.

We start by identifying the various types of messages flowing through the network. The ring has K nodes indexed by the set $\mathbf{K} = \{1,\ 2,\ \ldots\ K\}$. The type of a message is given by the ordered pair $m = (s, d) \in \mathbf{K} \times \mathbf{K}$, where s is its source node and d its destination one. For K nodes, we get $A = K^2$ distinct message types. Call $\mathbf{M}$ the set of all message types; by appropriately indexing them we can write $\mathbf{M} = \{M_1,\ M_2,\ \ldots\ M_a,\ \ldots\ M_A\}$.

Define now t_j^a ($j \in \mathcal{Z}$) to be the arrival time of the j^{th} message of type M_a to the ring. The interarrival times of type M_a messages are then given by $\tau_j^a = t_{j+1}^a - t_j^a \in \mathcal{R}_+$. Let also $l_j^a \in \mathcal{R}_+$ be the duration (time length) of the j^{th} message of type M_a to arrive to the ring. Given a fixed time origin, the first message of each type to arrive after time 0 is labeled 1, the second 2, and so on.

In order to transmit the j^{th} message of type $M_a = (s,\ d)$, which has source node s and destination node d, we must dedicate the sector of the ring from s to d (along the positive direction on the ring) for a length of time equal to l_j^a. Once transmission of a message has started, it cannot be interrupted (non-preemptive) until it is completed. If transmission of the j^{th} message is not possible at t_j^a, because the appropriate ring sector is not free (may even be partially occupied), then the message is queued up in a first-come-first-served queueing buffer Q_a of infinite size at the source node s. There are A buffers, one for each message type.

The traffic of arriving messages of type M_a is represented by the input random sequence

$$\mathbf{I}_a = \{(\tau_j^a,\ l_j^a),\ j \in \mathcal{Z}\},$$

for every $a \in \{1, 2, 3, \ldots, A\}$. The sequences $\mathbf{I}_a$ are collectively defined on a some probability space (Ω, F, P) and are assumed to be individually stationary and ergodic with respect the transformations θ_a,

$$\theta_a \mathbf{I}_a = \{(\tau_{j-1}^a, l_{j-1}^a), j \in \mathcal{Z}\},$$

Note that all the stochasticity of the system is essentially contained in the input traffic sequences $\{\mathbf{I}_a\}$.

The arrival rate of type M_a jobs on the input $\mathbf{I}_a$ is given by

$$\lambda^a = \frac{1}{E[\tau_1^a]} = \lim_{t\to\infty}\left[\frac{1}{t} \sum_{j \in \mathcal{Z}} \mathbf{1}\{t_j^a \in [0,\ t)\}\right],$$

(average number of type M_a arrivals per unit of time), where the limit exists because of the stationarity and ergodicity of $\mathbf{I}_a$. The total arrival rate is

$$\lambda = \sum_{a=1}^{A} \lambda^a,$$

and the arrival rate vector $\vec{\lambda}$ of the collection of input flows $\{\mathbf{I}_a, a \in \mathbf{A}\}$ is given by

$$\vec{\lambda} = (\lambda^1,\ \lambda^2,\ \dots\ \lambda^a,\ \dots\ \lambda^A)$$

Finally, the expected length the type M_a messages, that is, those carried by the input $\mathbf{I}_a$, is denoted by

$$p^a = E[l_1^a],$$

where $a \in \{1, 2, 3, \dots, A\}$, and the corresponding average message length vector by

$$\vec{p} = (p^1,\ p^2,\ p^3,\ \dots\ p^a,\ \dots\ p^A)$$

Given that t_1^a is the arrival time of the first M_a-type message to arrive after time 0, we start transmitting (processing) the inputs $\mathbf{I}_a$ $(a \in \mathbf{A})$ at time 0, assuming that the system has been empty before then. Obviously, there are many ways by which we could schedule (non-preemptively) the transmissions of messages on the ring. For example, we could have a global first-come-first-serve policy, where we transmit the messages one at a time and in the order of their arrival times; this is more or less what the token-ring architecture does, but obviously it is not efficient from a throughput point of view. Let $\mathbf{D}$ be the set of all possible non-preemptive dynamic transmission policies.

The problem of finding an optimal transmission policy (in the sense of achieving the highest possible throughput on the ring) is naturally raised. Closely related is the problem of identifying a policy that maximizes the stability region of the network with respect to the arrival rates vector $\vec{\lambda}$ (maximization is considered in the sense of set theoretic inclusion). Stability has to be expressed in some reasonable way, and finally the notion of rate-stability is employed (see section 3). The problems raised above are solved by constructing an distributed (on the nodes) transmission scheduling policy which achieves the set goals. Its rate-stability region turns out to be a superset of that of any other scheduling policy. We immediately proceed with the constructions and theorems, where the notions vaguely sketched above become well defined and clear.

2. A Distributed Static Transmission Schedule Maximizing the Throughput of the Ring

In this section we study a related problem, which is the *static* version of the previously described one. It turns out to be essential for the solution of the latter. The static problem is the following: given a large population of messages in the system (having been drawn consecutively from the input $\{\mathbf{I}_a\}$, ignoring their arrival times), we want to schedule them for transmission, so as to maximize the average number of messages transmitted per unit time (asymptotically, as their number tends to infinity); we construct here a distributed algorithm achieving this goal.

We start with the construction of the set $\mathbf{C}$ of combinations of message types which can be transmitted simultaneously on the ring, because their source-destination paths do not overlap. We refer to them as **consistent combinations**. $\mathbf{C}$ is the set of all A-dimensional vectors $\mathbf{c} = (c^1, c^2, \ldots c^a, \ldots c^A) \in \{0, 1\}^A$, such that we can concurrently transmit c^1 messages of type M_1, c^2 messages of type M_2, ... c^a messages of type M_a, ... and c^A messages of type M_A (recall $A = K^2$). Actually, $c^a = 0$ or 1, and by specifying the 1's we basically provide the combination of connections that can coexist on the ring without overlapping. Let B be the total number of consistent combinations; thus $\mathbf{C} = \{\mathbf{c}_1, \mathbf{c}_2, \ldots \mathbf{c}_b, \ldots \mathbf{c}_B\}$, where the $\mathbf{c}_b$'s are distinct consistent combinations. Segments of the ring may remain idle in some consistent combinations in $\mathbf{C}$.

Consider now the following *hypothetical problem:* suppose we have to transmit exactly *one* message from each message type in $\mathbf{M}$ and the time length of the message of type M_a is w^a; let then $\vec{w} = (w^1, w^2, \ldots w^a, \ldots w^A)$ be the length vector of different messages (one of each type). We assume that the messages can be transmitted *preemptively*, that is, we can arbitrarily time-slice them and transmit the slices separately and in any order (this is contrary to the situation in our original system, where transmissions are non-preemptive). The *objective* is to schedule the transmissions, so as to minimize the time to completion (makespan). The problem can be can be formulated as a linear programming problem in standard form, with variables $\vec{x} = (x^1, x^2, \ldots x^b, \ldots x^B) \in \mathcal{R}_+^B$, as follows:

minimize

$$\sum_{b=1}^{B} x^b = T(\vec{x})$$

subject to

$$\sum_{b=1}^{B} x^b c_b^a = w^a, \quad \text{for every} \quad a \in \{1, 2, \ldots A\},$$

$$x^b \geq 0, \quad \text{for every} \quad b \in \{1, 2, \ldots B\}.$$

Due to its structure, the program has an optimal feasible solution (although it may not be unique). Let now

$$\vec{x}_o(\vec{w}) = (x_o^1(\vec{w}), x_o^2(\vec{w}), \ldots x_o^b(\vec{w}), \ldots x_o^B(\vec{w})) \in \mathcal{R}_+^B$$

be a *basic optimal feasible solution* of the linear program (see [6]), and define

$$\sum_{b=1}^{B} x_o^b(\vec{w}) = T_o(\vec{w}).$$

Thus, if we preemptively transmit the messages for $x_o^1(\vec{w})$ time units under the combination $\mathbf{c}_1$, then for $x_o^2(\vec{w})$ time units under $\mathbf{c}_2$, ... and finally for $x_o^B(\vec{w})$ time units under $\mathbf{c}_B$, we finish the transmissions of all the messages in the minimum possible time $T_o(\vec{w})$. Note also that since $\vec{x}_o(\vec{w})$ is a basic optimal feasible solution, only A out of the B (usually much larger than A) x_o's are non-zero, and thus only A combinations have to be actually tried.

Based on the previous constructions, we now define some statistical parameters of the system. First let

$$\pi^a = \frac{\lambda^a}{\lambda} p^a,$$

for $a \in \{1,2,3,...,A\}$, where $p^a = E[l_1^a]$, and

$$\vec{\pi} = (\pi^1, \pi^2, ... \pi^a, ... \pi^A).$$

We assume that $0 < p^a < \infty$ and $0 < \lambda^a < \infty$ for all a's. Now, define the *basic* optimal feasible solution (see [6])

$$\vec{x}_o(\vec{\pi}) = (x_o^1(\vec{\pi}),\ x_o^2(\vec{\pi}),\ ...\ x_o^b(\vec{\pi}),\ ...\ x_o^B(\vec{\pi}))$$

and

$$\bar{T} = T_o(\vec{\pi}) = \sum_{b=1}^{B} x_o^b(\vec{\pi}),$$

where we have substituted for the workloads $\vec{w} = (w^1,\ w^2, ...,\ w^a,\ ...\ w^A)$ the values $\vec{\pi} = (\pi^1,\ \pi^2,\ ...\ \pi^a,\ ...\ \pi^A)$ in the previously constructed linear program. Finally, let $\mathbf{C}^{basic}(\vec{\pi}) \subset \mathbf{C}$ be the set of consistent combinations $\mathbf{c}_b$ corresponding the non-zero entries of the basic optimal feasible solution $\vec{x}_o(\vec{\pi})$; we call them *basic* consistent combinations and there are at most $A = K^2$ of them (see [6]). Obviously, $\bar{T} = T_o(\vec{\pi})$ and $\mathbf{C}^{basic}(\vec{\pi})$ are statistical parameters of the input $\{\mathbf{I}_a\}$ (being functions of $\vec{\pi}$) and have to be computed (once and off-line) by solving the linear program with entries $\vec{\pi}$.

For any given sample path of the collection of inputs $\{\mathbf{I}_a\}$, define now the random time z_N to be the arrival time of the N^{th} message (of any type) to arrive after time z on this sample path. That is, there are N messages arriving in the time interval $[z, z_N]$. The number of messages of type M_a arriving in $[z, z_N]$ is

$$\Delta_N^a(z) = \sum_{j \in Z} \mathbf{1}\{t_j^a \in [z, z_N]\}.$$

We isolate all these N messages from the sample path and place them in the system buffers. We start processing them only after all of them have been gathered. We are interested in constructing a *non-preemptive* schedule that minimizes the average transmission time per message, asymptotically, as the total number N of messages tends to infinity (that is, maximizes the throughput). This is the *static* version of the problem we have been talking about, since we ignore the arrival times of the messages.

The total transmission time of messages of type M_a among the N messages defined above is $L_N^a(z) = \sum_{j \in Z} l_j^a \mathbf{1}\{t_j^a \in [z, z_N]\}$, for $a \in \mathbf{A}$. If we were allowed to transmit the messages *preemptively* (time-slice them), we could minimize their total processing time by solving the linear program (previously constructed for the hypothetical problem), with $\vec{w}$ equal now to

$$\vec{L}_N(z) = (L_N^1(z),\ L_N^2(z),\ ...\ L_N^a(z),\ ...\ L_N^A(z)).$$

The minimal total transmission time would then be $T_o(\vec{L}_N(z))$. As N tends to infinity, we get

$$\lim_{N \to \infty} [T_o(\frac{\vec{L}_N(z))}{N}] = T_o(\lim_{N \to \infty} [\frac{\vec{L}_N(z)}{N}]) = T_o(\vec{\pi}) = \bar{T},$$

for any s, almost surely, because of the linearity and continuity of the operations involved in computing the function $T_o(\ )$ in the linear program. In view of the above, we can conclude that $1/\bar{T}$ is the **maximal static throughput** of the ring, achieved when can process the messages preemptively (time-slice them).

However, in our system, the messages have to be transmitted non-preemptively, and the schedule has to observe this restriction. Thus, we construct a distributed *non-preemptive* schedule Π_d, which is shown (Theorem 1) to also achieve the maximal throughput $1/\bar{T}$, thus being asymptotically optimal as $N \to \infty$.

Algorithm I: *Static Distributed Transmission Schedule* Π_d.

The algorithm uses a *control token* which is passed around the nodes. The node holding the token has temporary control of the network (in the sense explained below) and is transmitting. However, other nodes may also transmit concurrently (*contrary* to the standard token-ring architecture, where only the node holding the token can transmit). Let $\mathbf{C}_k^{basic}$ be the set of all *basic* consistent combinations $\mathbf{c}_b \in \mathbf{C}^{basic}(\vec{\pi})$ in which node-k ($k \in \{1, 2, 3, ..., K\}$) is transmitting (is active, transmitting on its outgoing link and receiving on its incoming one). Let also $\mathbf{\Gamma}$ be a set-variable with values that are subsets of $\mathbf{C}$ (that is, $\mathbf{\Gamma} \subset \mathbf{C}$). The algorithm has a main part and a tail one.

A) MAIN PART:

Step 0: The control token is initially given to an arbitrarily chosen node, say, node-1. Also, we initially set $\mathbf{\Gamma} = \mathbf{C}^{basic}(\vec{\pi})$.

Step 1: Given that node-k ($k \in \{1, 2, 3, ..., K\}$) has the control token:

Step 1.1: Node-k arbitrarily chooses a basic consistent combination $\mathbf{c}_b \in \mathbf{C}_k^{basic} \bigcap \mathbf{\Gamma}$.

Step 1.2: Node-k composes and transmits a control packet (message) around the ring (in the positive direction) to set its connectivity (configuration) to the combination $\mathbf{c}_b$. The control packet travels from every node to the next one on the ring. When it reaches, say, node-k', it *scans* it. That is, it waits until the message currently being transmitted from node-k completes transmission, before blocking transmission of the following messages of the previously active type (if any), and then setting node-k' to be consistent with the new configuration $\mathbf{c}_b$ (enabling - or not - the transmission of messages of certain type from k' under $\mathbf{c}_b$). It then moves on to the next node $k' + 1$ to perform the same scanning function. Thus, it travels around the ring, eventually returning to node-k (where it started from), where it is now absorbed and destroyed, having configured the whole ring under the chosen combination $\mathbf{c}_b$. After the absorption of the control packet node-k proceeds with the following Step 1.3.

Step 1.3: Assuming that node-k is now configured (under $\mathbf{c}_b$) to transmit messages of type $M_a \in \mathbf{M}$, it proceeds with the consecutive (serial) transmissions of the first

$$\lfloor \frac{x_0^b(\vec{\pi})}{\pi^a} \times \Delta_N^a(z) \rfloor$$

messages found residing in the queueing buffer Q_a (assuming there are any; otherwise continue with Step 2.4). $\lfloor\ \rfloor$ denotes the integer part of the argument. During the random time interval when node-k transmits the above specified number of messages from Q_a, the other nodes on the network which are enabled under $\mathbf{c}_b$ to transmit, do concurrently transmit consecutive messages from the buffers specified by $\mathbf{c}_b$ (hence the spatial reuse of the ring). When the last of the previously specified messages from Q_a completes transmission the algorithm proceeds to the next Step 1.4.

Step 1.4: Upon completion of Step 1.3, combination $\mathbf{c}_b$ is considered exhausted and is removed from the set $\mathbf{\Gamma}$, that is,

$$\mathbf{\Gamma} \leftarrow \mathbf{\Gamma} - \{\mathbf{c_b}\}$$

Step 1.5: If $\mathbf{C}_k^{basic} \bigcap \mathbf{\Gamma} \neq \emptyset$ (there are still basic combinations under which node-k can transmit), go to *STEP 1.1.* Otherwise, if $\mathbf{C}_k^{basic} \bigcap \mathbf{\Gamma} = \emptyset$, go to *STEP 2.*

Step 2: If $k \neq K$, pass the token to the next node on the ring, that is,

$$k \leftarrow k+1,$$

and go to STEP 1.

If $k = K$, *EXIT* the main part and go to *TAIL PART.*

B) TAIL PART:

By the end of the main part of the algorithm, some jobs will have been left in the buffers. We can then go around the ring once more, completely emptying the non-empty queues individually (in the worst case). It turns out that this forms a tail operation that has no effect on the throughput of the system, as the number of messages becomes very large (see the sketch of proof of Theorem 1). This completes the descriptions of Algorithm I. □

Basically, the control token traveling around the ring, reconfigures the connectivities of the nodes, passing through all the *basic* configurations (basic consistent combinations) in $\mathbf{C}^{basic}(\bar{\pi})$. Thus the algorithm will exit its main part after it loops around it for at most $A = k^2$ times (no matter what N is). While in each combination, it processes the appropriate messages in a balanced way (so as to emulate - asymptotically - the processing pattern used by the optimal preemptive schedule).

Theorem 1: *Asymptotic Optimality of Schedule* Π_d.

Let $T_N^{\Pi_d}(z)$ be the time to process the N jobs in $[z, z_N]$ For all stationary and ergodic sequences $\{\mathbf{I}_a\}$, we have

$$\lim_{N\to\infty}[\frac{T_N^{\Pi_d}(z)}{N}] = \lim_{N\to\infty}[\frac{E[T_N^{\Pi_d}(z)]}{N}] = T_o(\bar{\pi}) = \bar{T}$$

almost surely. Thus, the non-preemptive schedule Π_d achieves the maximal throughput $1/\bar{T}$, and in that sense it is asymptotically optimal (clearly it is suboptimal for small N's).

Main Idea of Proof: Although the technicalities of the proof are quite involved, the key aspects can be intuitively captured. Intuitively speaking, Π_d tries to emulate the transmission pattern of the optimal preemptive schedule, which we get by solving the linear program. However, the "discretization" enforced by the fact that Π_d is non-preemptive generates some "boundary effects" at the times that the configuration of the ring switches to a new basic combination. For small N's these "boundary irregularities" differentiate significantly the two transmission patterns. However, as $N \to \infty$, the irregularities are smoothed out (on average) and two patterns look more an more alike. Divided by $N \to \infty$ they completely die out (this is what happens to the tail part of the algorithm) and the two schemes achieve the same throughput. $\square$

3. A Distributed Dynamic Transmission Schedule Maximizing the Rate-Stability Region of the Ring

We now investigate the original system, where messages arrive dynamically at random times and the network operates continuously. We construct a non-preemptive distributed dynamic transmission schedule $D_{\Pi_d} \in \mathbf{D}$, which is shown to maximize the rate-stability region of the network, that is, the set of $\vec{\lambda}$'s such that the departure rates of the messages of various types are equal to their arrival rates (given that the mean message lengths $\vec{p}$ are constant). D_{Π_d} utilizes Π_d, adapting it to the dynamic environment. D_{Π_d} operates on any given sample path of the input $\{\mathbf{I}_a\}$, according to the following algorithm.

Algorithm II: *Dynamic Distributed Transmission Schedule* D_{Π_d}.

Like Algorithm I, this algorithm too uses a *control token* which is passed around the nodes allowing the one holding it to transmit and giving it temporary control of the network (in the sense explained below). Again, however, other nodes may also transmit concurrently (*contrary* to the standard token-ring architecture, where only the node holding the token can transmit). There is naturally some delay associated with passing the control token from one node to the next. Recall that $\mathbf{C}_k^{basic}$ is the set of all *basic* consistent combinations $\mathbf{c}_b \in \mathbf{C}^{basic}(\vec{\pi})$ in which node-k ($k \in \{1,2,3,...,K\}$) is transmitting (is active, transmitting on its outgoing link and receiving on its incoming one). Again, let $\mathbf{\Gamma}$ be a set-variable with values that are subsets of $\mathbf{C}$ ($\mathbf{\Gamma} \subset \mathbf{C}$). Let also $\Delta^a, a \in \{1,2,3,...A\}$ be integer valued variables (will denote the populations of type M_a messages in the network below). Algorithm II is actually an extension of Algorithm I in a dynamic setup and works as follows.

Step 0: We start operating the network at time 0 on a sample path of the random inputs $\{\mathbf{I}_a\}$, assuming it is empty at time 0. The control token is initially given to an arbitrarily chosen node, say, node-1. Also, we initially set $\mathbf{\Gamma} = \mathbf{C}^{basic}(\vec{\pi})$ and $\Delta^a = 0$ for every $a \in \{1,2,3,...,A\}$.

Step 1: Given that node-k ($k \in \{1,2,3,...,K\}$) has the control token:

Step 1.1: Node-k arbitrarily chooses a basic consistent combination $\mathbf{c}_b \in \mathbf{C}_k^{basic} \bigcap \mathbf{\Gamma}$.

Step 1.2: Node-k composes and transmits a control packet (message) around the ring (in the positive direction) to set its connectivity (configuration) to the combination $\mathbf{c}_b$. The control packet travels from every node to the next one on the ring. When it reaches, say, node-k', it *scans* it. That is, it waits until the message currently being transmitted from node-k completes transmission, before blocking transmission of the following messages of the previously active type (if any), and then setting node-k' to be consistent with the new configuration $\mathbf{c}_b$ (enabling - or not - the transmission of messages of certain type from k' under $\mathbf{c}_b$). It then moves on to the next node $k'+1$ to perform the same scanning function. Thus, it travels around the ring, eventually returning to node-k (where it started from), where it is now absorbed and destroyed, having configured the whole ring under the chosen combination $\mathbf{c}_b$. After the absorption of the control packet node-k proceeds with the following Step 1.3.

Step 1.3: Assuming that node-k is now configured (under $\mathbf{c}_b$) to transmit messages of type $M_a \in \mathbf{M}$, it proceeds with the consecutive (serial) transmissions of the first

$$\lfloor \frac{x_0^b(\vec{\pi})}{\pi^a} \times \Delta^a \rfloor$$

messages found residing in the queueing buffer Q_a (assuming there are any; otherwise continue with Step 1.4). $\lfloor\ \rfloor$ denotes the integer part of the argument. (The variables Δ^a are updated in Step 3.) During the random time interval when node-k transmits the above specified number of messages from Q_a, the other nodes on the network which are enabled under $\mathbf{c}_b$ to transmit, do concurrently transmit consecutive messages from the buffers specified by $\mathbf{c}_b$ (hence the spatial reuse of the ring). When the last of the previously specified messages from Q_a completes transmission the algorithm proceeds to the next Step 1.4.

Step 1.4: Upon completion of Step 1.3, combination $\mathbf{c}_b$ is considered exhausted and is removed from the set $\mathbf{\Gamma}$, that is,

$$\mathbf{\Gamma} \leftarrow \mathbf{\Gamma} - \{\mathbf{c_b}\}$$

Step 1.5: If $\mathbf{C}_k^{basic} \bigcap \mathbf{\Gamma} \neq \emptyset$ (there are still basic combinations under which node-k can transmit), go to *STEP 1.1.* Otherwise, if $\mathbf{C}_k^{basic} \bigcap \mathbf{\Gamma} = \emptyset$, go to *STEP 2.*

Step 2: If $k \neq K$, pass the token to the next node on the ring, that is,

$$k \leftarrow k + 1,$$

and go to *STEP 1.* If $k = K$, go to Step 3.

Step 3: Reset $k = 1$ and $\mathbf{\Gamma} = \mathbf{C}^{basic}(\vec{\pi})$. Update the Δ^a's by setting them equal to the current populations of messages of corresponding types M_a in the network. That is,

$$\begin{aligned} \Delta^a \quad \leftarrow \quad & \text{number of messages of type } M_a \text{ residing in the buffer } Q_a \\ & \text{when Algorithm II enters Step 3} \end{aligned}$$

Go to *STEP 1.*

This completes the description of Algorithm II. □

Step 1 of algorithm II is structurally identical to Step 1 of algorithm I. Step 2 of alg. II differs from Step 2 of alg. I only to the extend that it eventually leads to Step 3 of alg. II, which restarts the whole process of Steps 1 & 2. Thus, algorithm II loops through its Steps 1,2,3 for ever, as it is processing continuously the incoming messages.

To quantify the notion of instability in the following theorem, we now define the *completion time* c_t^D to be the time for the network to become empty, given that no message is accepted for transmission after time t and the system operates under the dynamic schedule $D \in \mathbf{D}$. That is, we process the inputs $\{\mathbf{I}_a\}$ in the interval $[0, t)$ under $D \in \mathbf{D}$

and then stop accepting any more jobs in the buffers. Then, the messages already in the system will all be transmitted by the time $t + c_t^D$.

Let us now define $y_j^a(D)$, $j \in \mathcal{Z}$, $D \in \mathbf{D}$ to be the *departure time* of the j^{th} job of type M_a, given that the ring operates under the dynamic transmission schedule $D \in \mathbf{D}$. We can then define the **departure rates**

$$\mu_D^a = \lim_{t \to \infty} [\frac{1}{t} \sum_{j \in \mathcal{Z}} \mathbf{1}\{z_j^a(D) \in [0,\ t)\}],$$

$a \in \{1, 2, 3, ..., A\}$, provided that the limits exist and are constant almost surely.

Recall that $\vec{\lambda} = (\lambda^1, \lambda^2, ...\lambda^a, ...\lambda^A)$, $\lambda = \sum_{a=1}^{A} \lambda^a$ (total arrival rate), and $\vec{p} = \{p^1,\ p^2,\ ...\ p^a,\ ...\ p^A\}$, (where $p^a = E[l_1^a]$). We can then define the sets:

$$\mathbf{S}(\vec{p}) = \{\vec{\lambda} \in \mathcal{R}_+^A : T_o(\ (\lambda^1 p^1, \lambda^2 p^2, ...\lambda^a p^a, ...\lambda^A p^A)\ ) < 1\} = \{\vec{\lambda} \in \mathcal{R}_+^A : \lambda < \frac{1}{T_o(\vec{\pi})}\},$$

and

$$\mathbf{S}_b(\vec{p}) = \{\vec{\lambda} \in \mathcal{R}_+^A : T_o(\ (\lambda^1 p^1, \lambda^2 p^2, ...\lambda^a p^a, ...\lambda^A p^A)\ ) = 1\ \} = \{\vec{\lambda} \in \mathcal{R}_+^A : \lambda = \frac{1}{T_o(\vec{\pi})}\}$$

(noting that $\lambda^a p^a = \lambda \pi^a$ for all a's), which depend on the average message lengths $\vec{p}$. Because of the structure of $T_o(\ )$ (see section 2) the set $\mathbf{S}(\vec{p})$ is a *convex finite polyhedron* in $\mathcal{R}_+^A$ and $\mathbf{S}_b(\vec{p})$ is a subset of the boundary of this polyhedron. Let finally

$$\mathbf{R}(D, \vec{p}) = \{\vec{\lambda} \in \mathcal{R}_+^A :\ \mu_D^a = \lambda^a \ \text{ for every a} \in \{1, 2, 3, ...A\}\}$$

be the *rate-stability region* of the network operating under dynamic schedule $D \in \mathbf{D}$, that is, the set of arrival rates $\vec{\lambda}$ for which the departures rates are well defined and equal to the arrival rates for the various message types (given fixed message length statistics $\vec{p}$). The following theorem shows that $\mathbf{S}(\vec{p})$ is the maximal rate-stability region of the ring network.

Theorem 2: D_{Π_d} *maximizes the rate-stability region of the ring network*

1) **Instability:** If $\vec{\lambda} \in \mathcal{R}_+^A - \{\mathbf{S}(\vec{p}) \cup \mathbf{S}_b(\vec{p})\}$, then for *every* dynamic transmission schedule $D \in \mathbf{D}$, we have

$$\lim_{t \to \infty} [c_t^D] \ = \ + \infty$$

and

$$\text{liminf}_{t \to \infty}\ [\frac{1}{t} \sum_{j \in \mathcal{Z}} \mathbf{1}\{y_j^a(D) \in [0,\ t)\}] < \lambda^a,$$

for at least one $a \in \{1, 2, 3, ..., A\}$, and the network can be characterized as *unstable.*

2) **Rate-Stability:** If $\vec{\lambda} \in \mathbf{S}(\vec{p})$ and the ring network operates under the dynamic transmission schedule $D_{\Pi_d} \in \mathbf{D}$, we get

$$\lim_{t \to \infty}\ [\frac{1}{t} \sum_{j \in \mathcal{Z}} \mathbf{1}\{y_j^a(D_{\Pi_d}) \in [0,\ t)\}] \ = \ \mu_{D_{\Pi_d}}^a \ = \ \lambda^a,$$

for every $a \in \{1, 2, 3, ..., A\}$, almost surely. That is, the departure rates of the various types of jobs are exactly equal to their arrival rates, and the ring network can be characterized as *rate-stable.* Therefore, *excluding* points on the boundary $\mathbf{S}_b(\vec{p})$ which can potentially lead to critical (anomalous) stability behavior, we have

$$\mathbf{R}(D, \vec{p}) \subseteq \mathbf{R}(D_{\Pi_d}, \vec{p}) = \mathbf{S}(\vec{p}), \text{ for every D} \in \mathbf{D},$$

and D_{Π_d} maximizes the rate-stability region of the network (under fixed $\vec{p}$).

Main Idea of Proof: Despite the actual complications of the proof, the key observation for the case of rate-stability is quite intuitive. Every time that D_{Π_d} visits its Step 3, the Δ^a's register the current populations (numbers) of messages of all types M^a. If the message populations during some visit are small, Steps 1 & 2 of D_{Π_d} will potentially result in far from optimal processing; this will tend to increase the population sizes encountered in the immediately following visits to Step 3. However, the larger the populations $\{\Delta^a\}$ become, the better processed they are by Steps 1 & 2 of D_{Π_d} (due to the asymptotic optimality of of Π_d, which is basically implemented in Steps 1 & 2 of D_{Π_d}); this tends to *decrease* the population sizes during subsequent visits to Step 3. The latter tendency works towards not letting the population sizes $\{\Delta^a\}$ drift towards infinity (at least not superlinearly with respect to time) and this leads to rate-stability. □

We conclude that the transmission schedule D_{Π_d} rate-stabilizes the network for every $\vec{\lambda} \in \mathbf{S}(\vec{p})$, while the network is unstable under *all* possible $D \in \mathbf{D}$ if $\vec{\lambda}$ does not belong to the set $\mathbf{S}(\vec{p}) \cup \mathbf{S}_b(\vec{p})$.

4. Final Remarks

In order to observe the page limit on proceedings papers, we have made a very brief presentation of the issues and results, giving basically a skeleton of the work. However, there are many other important issues that have not been discussed here. We just mention some of them below.

The algorithms I & II can be relaxed, retaining their optimality properties. For example, in Step 1.2, as the control packet travels around the ring, communication paths that have already been formed can start being used, without waiting for the whole ring to switch to the new configuration. Other distributed versions of the algorithms have also been investigated (appropriately choosing the consistent combinations so as to improve access fairness for the users, pipelining messages on active and emerging communication paths etc.)

Important issues of performance evaluation have also been studied analytically and/or by simulation (blocking probabilities under finite buffer sizes, statistics of intervisit times to Step 3 of alg. II, average message delays etc.)

Recall that the constructions presented in section 2 are based on the solution of a linear program, and the identification of a basic optimal feasible solution and the appropriate basic consistent combinations. The solution of the linear program may not be easy for certain values of $\vec{\pi}$. In that case we can search for simple *near-optimal* solutions and appropriately select a group of consistent combinations to configure the network. Algorithms I & II can be modified to operate in such a case, but will naturally lead to only near-optimal results.

Finally, observe that the operation of the ring is based on dynamically switching between various consistent combinations, so that the ring is reconfigured again and again in order to satisfy the multiple communication requests. We can view this reconfiguration as happening by a *metatoken* being exchanged between the various consistent combinations. The one holding the metatoken can exclusively configure the ring and serve the communication requests that are compatible with it. It sustains the configuration until it releases the metatoken to another consistent combination. In the token-ring protocol a token circulates among the nodes and the one holding it can exclusively transmit. In our situation here, a metatoken circulates among various consistent combinations and the one holding it can exclusively configure the ring (which is now spatially reused at the level of nodes). Therefore, we call the operational scheme proposed in this paper *MetaToken-Ring Protocol.*

5. References

[1] N. Bambos and A. Nguyen (1992). "Optimal Message Flow in Ring Networks with Spatial Reuse," Proceedings of the 31st IEEE Conference on Decision and Control, Tuscon, AZ, Dec. 1992.

[2] N. Bambos and J. Walrand (1993). "Scheduling and Stability Aspects of a General Class of Parallel Processing Systems." Advances in Applied Probability, 25, 176-202, 1993.

[3] I. Cidon and Y. Ofek (1990). "Metaring - A Full-Duplex Ring with Fairness and Spatial Reuse", in Proceedings of IEEE INFOCOM '90, San Francisco, CA, pp. 969-981, June 1990.

[4] Durrett, R. (1991). Probability: Theory And Examples. Wadsworth & Brooks/Cole.

[5] I. Kessler, A. Krishna (1991). "Analysis of a Fair, Slotted Ring with Spatial Reuse", Proceedings of the 29th Allerton Conference, IL.

[6] D. Luenberger "Linear and Nonlinear Programming" Addison-Wesley, Reading, MA, 1984.

Operation of Non-Bus-Oriented Single-Hop Interconnections: Busy-Tone Multiple Access and Folded Schedules *

Yitzhak Birk and Noam Bloch
Electrical Engineering Department
Technion – Israel Institute of Technology
Haifa 32000, Israel

Abstract

The use of transmissive star couplers permits the construction of switchless, non-bus-oriented single-hop interconnections. By equipping each station with a small, fixed number of transmitters and receivers, such interconnections whose uniform-traffic capacity increases with the number of stations can be constructed and can be laid out efficiently. However, the fixed round-robin schedules used to attain the high capacity result in prohibitive delays at low loads, and throughput is very sensitive to skews in traffic load. In this paper, two techniques for mitigating these problems are presented along with an initial assessment of their impact and limitations.

Key words and phrases: *fiber optic LANs, shared directional multichannel, multiple access, folded schedule, busy-tone multiple-access, local area networks.*

1. Introduction

Single-hop interconnections, SHI's for short, are static, switchless interconnections which provide a (possibly shared) path among any pair of stations at all times. The most prominent such interconnection is the bus, which is used for LANs such as Ethernet. While flexible and simple, the single bus permits at most one ongoing transmission and is thus a bottleneck.

One way of solving this problem is giving up the single-hop connectivity or adding switches to dynamically partition the network and route concurrent messages to their respective destinations without conflict. This approach has been employed in the telephone network as well as in wide-area networks. More recently, it is also being applied to LANs in the form of switching hubs. However, this comes at a cost.

In the last several years, various researchers have explored ways of permitting concurrent transmissions while retaining static, single-hop connectivity through a passive fabric

---

*The work was supported in part by a grant from Intel corporation, and in part by the M & C Papo Research Fund through the Technion VP of Research. email: birk@ee.technion.ac.il

[1][2][3][4][5][6] [7][8][9][10][11]. Advantages include simplicity, as well as the ability to permit each (source, destination) pair to communicate at a different rate, depending on their equipment. This is particularly attractive in the fiber-optic domain. To permit concurrent transmissions while retaining static, single-hop connectivity, one must either use tunable components or equip each station with multiple transmitters and/or receivers. The discussion in this paper is limited to non-tunable components.

Initial results involved the use of "bus-oriented" interconnections. These can be described as a collection of "buses", such that any given transmitter or receiver is connected to a single bus, and for any two stations there is at least one bus to which they are both connected. If each station is equipped with c_T transmitters and c_R receivers, it was shown that up to $c_T \cdot c_R$ equally populated buses can be constructed, such that any two stations have a bus in common [3][4]. If $c_T = c_R = c$ and a station must connect its transmitters and receivers to the same buses, the maximum number of buses is slightly lower: $c^2 - c + 1$. The number of stations, N, is assumed to be greater than c^2 [12].

The use of transmissive star couplers permits the construction of "non-bus-oriented" SHI's. With this *shared directional multichannel* [6], unlike with buses, the sets of receivers reachable from any two transmitters needn't be identical or disjoint. This additional degree of freedom has permitted the construction of SHI's whose uniform-traffic capacity increases polylogarithmically with the number of stations. For example, if 160 stations are each equipped with two transmitters and two receivers, an SHI that permits 25 concurrent transmissions can be constructed [6]. This is a dramatic improvement over the highest capacity attainable with bus-oriented interconnections, 4 in this example. Moreover, the high-concurrency SHI's can be laid out with only a fixed number of fiber segments per station, and with a path loss of only N (the number of stations) [13][14][15]. While our focus is on the case of a small, fixed number of transmitters and receivers per station, it has recently been shown in [9] that concurrency N can be attained with $N^{1/3}$ transmitters and receivers per station. This was achieved by extending a construction in [6] and [8] for $N = 8$, and then repeatedly applying a composition described in [6]. A similar result can be obtained by simply composing two small designs that were described in [6] and [8].

Unfortunately, the high degree of concurrency of SHI's was demonstrated only using a fixed round-robin transmission schedule. Worse yet, each schedule slot is allocated to specific (source, destination) pairs. This is in contrast with conventional TDMA, wherein the right to transmit is given to specific sources with no restriction on the destination. With N stations, the length of the schedule round is therefore N^2 divided by the degree of concurrency. This causes undesirably high low-load delay (half of a very long round on average), and extreme sensitivity of throughput to traffic pattern. For example, the schedule round for an interconnection among N stations, each with two transmitters and a single receiver, would consist of $N^2/\log_2 N$ time slots: in each time slot, $\log_2 N$ specific (source,destination) pairs would be allowed to communicate.

In this paper, we begin to address these problems by exploring other ways of operating the interconnections. In section 2, we introduce our notation and briefly describe an SHI with capacity $\log_2 N$. In section 3, we consider "random access" techniques and derive some bounds. In section 4, we introduce the technique of "schedule folding", and show the extent to which it can be applied. Section 5 offers conclusions and directions for future research.

2. Specification of an SHI: wiring and schedule

Consider a set $S = \{s_1, \ldots, s_{N_S}\}$ of N_S source stations (SS's) communicating with a set $D = \{d_1, \ldots, d_{N_D}\}$ of N_D destination stations (DS's) in the following way: each station in S has c_T transmitters which we index by a set T, $|T| = c_T$, and each station in D has c_R receivers, indexed by R, $|R| = c_R$.
We will restrict the discussion here to SHI's that provide a single path between each SS and every DS. Thus, for each pair of stations $s \in S$ and $d \in D$, exactly one transmitter of s is connected to exactly one receiver of d. Denote the index of that transmitter by $W_1(s,d) \in T$, and the index of that receiver by $W_2(s,d) \in R$. Let W be the $N_S \times N_D$ matrix indexed by $S \times D$, whose entries are $W(s,d) = (W_1(s,d), W_2(s,d))$. W is called the *Wiring matrix.* For convenience, we will refer to an SHI by its size $(c_T, c_R; N_S, N_D)$ whenever there is no possibility of confusion.

Messages are transmitted at discrete time slots, and are all one slot in duration. The reception rules are those of a *shared directional multichannel* [6], namely: a message is received successfully by a DS iff that DS is the addressee and the receiver that hears the transmission hears no other transmissions in the same time slot (no collision).

A round-robin *Transmission Schedule* is an $N_S \times N_D$ matrix X indexed by $S \times D$ whose entries $X(s,d)$ have time-slots as values. A schedule X is *compatible* with a wiring W if and only if for all $(s,d) \in S \times D$, s communicates successfully with d at time $X(s,d)$. Formally, a schedule X is compatible with a wiring W iff for any two pairs $(s_1, d_1) \neq (s_2, d_2) \in S \times D$ the following holds: if both $W_1(s_1,d_1) = W_1(s_1,d_2)$ and $W_2(s_1,d_2) = W_2(s_2,d_2)$, then $X(s_1,d_1) \neq X(s_2,d_2)$. Fig. 1 illustrates the compatibility rules.

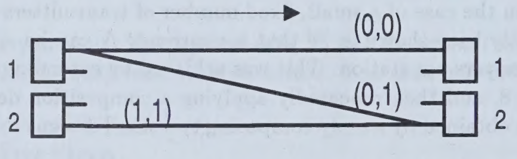


Figure 1: Compatibility of wiring and schedule: since $W_1(1,1) = W_1(1,2)$ $(= 0)$ and $W_2(1,2) = W_2(2,2)$ $(= 1)$, transmissions from SS_1 to DS_1 and from SS_2 to DS_2 must not be scheduled for the same slot. (Collision at DS_2.)

Finally, $f(c_T, c_R; N_S, N_D)$ denotes the minimum number of different time slots for which there exist compatible wiring W and transmission schedule X (permitting one message from each SS to every DS), and note that $f(c_T, c_R; N_S, N_D) = f(c_R, c_T; N_D, N_S)$ [6].

Example: An SHI with capacity $\log_2 N$[6]
Consider k SS's, each with two transmitters, and 2^k DS's, each with a single receiver. Let each SS be labeled with a k-bit vector containing a single '1', and each DS with a k-bit vector. Let $W(s,d) = (s \cdot d, 0)$. (Inner product of the two k-bit vectors, modulo 2; $W_2(s,d) \equiv 0$ because $c_R = 1$.) Finally, let $X(s,d) = s + d$. (Bit-by-bit addition, modulo 2.) Thus, each SS partitions the DS's based on a different bit in their label. Fig. 2 depicts such an SHI for $k = 4$. (Refer only to the top "layer" of SS's.)

This interconnection allows one message to be sent from each SS to each DS at a rate of k per slot [6]. As depicted in Fig. 2, this interconnection can be augmented to form a $(2, 1; 2^k, 2^k)$ SHI. We note in passing that a modification that permits one additional concurrent transmission was presented in [8].

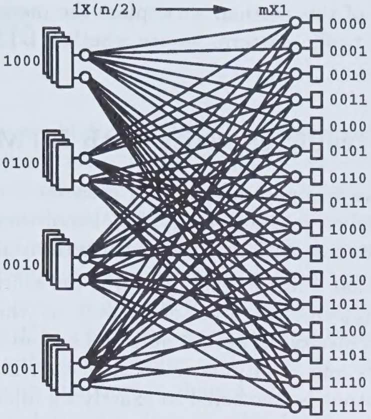


Figure 2: An interconnection of 2^k 2-transmitter SS's and 2^k single-receiver DS's. ($k = 4$.) The SS's are of k "types", each wired differently, with $2^k/k$ identically-wired SS's in each type. Rectangles and circles denote stations and star couplers, respectively.

3. Random-access schemes

Random access schemes such as ALOHA and CSMA-CD are attractive, since they help allocate the transmission media in an adaptive way, and sharply reduce low-load delay. In this section, we address the question of whether such schemes can also attain the high degree of concurrency possible with round-robin schedules at heavy, uniform loads.

3.1 Candidate access schemes

CSMA and CSMA-CD cannot be used with non-bus-oriented SHI's, since collisions are only defined at individual receivers, which hear different subsets of transmitters.

With slotted ALOHA, the maximum uniform-traffic throughput attainable by any single-path SHI is $\frac{1}{e} \cdot c_T \cdot c_R$, so the advantage of non-bus-oriented SHI's is lost [3][6]. Slotted ALOHA can, nevertheless, be used at very low loads to obtain low delay or adapt to skews in the traffic pattern.

Busy-tone multiple-access, BTMA for short [16], was originally invented as a solution to the "hidden terminal" problem in packet radio: station A can hear B and C and they can hear it, but a mountain prevents B and C from hearing each other, so they may cause collisions at A. The proposed solution was to have the receiving station transmit a busy tone on a side channel, and a station hearing a busy tone would not transmit.

BTMA is directly applicable to non-bus-oriented SHI's, since "hidden terminal" situations are abundant by design. To implement this scheme, each receiver would be equipped with a small busy-tone transmitter that transmits along the same fiber in the reverse direction. Similarly, each transmitter would be equipped with a busy-tone sensor. The busy tone emitted by a receiver reaches exactly those transmitters which, by transmitting, could cause a destructive collision at that receiver. To avoid unnecessary blocking of transmitters, this should be done only by receivers that are receiving a packet addressed to them.

In the remainder of this section, we explore the merits of BTMA, focusing on heavy load and a uniform traffic pattern, to see whether BTMA should be used exclusively across the load range.

3.2 Uniform-traffic capacity with BTMA

We begin by deriving an upper bound, and then show that it is tight. The derivations will be made for a more general setting of a shared directional multichannel, and then converted to SHI's among stations with multiple transmitters and receivers.

A shared directional multichannel [6], SDM for short, comprises a set of inputs, to which we connect transmitters, a set of outputs to which we connect receivers, and a specification of connectivity between inputs and outputs. For our purposes, we consider SDM's with t inputs and r outputs, such that each input is connected to exactly d_T outputs and each output is connected to exactly d_R inputs. Clearly, $t \cdot d_T \equiv r \cdot d_R$.

To obtain an upper bound, we use an "idealized" BTMA: zero propagation delay and no race conditions. In each time slot, we order the transmitters that have a packet for transmission at random, and begin to execute BTMA in the chosen order. Each transmitter is assumed to pick a random destination, and the choices are assumed to be independent from transmitter to transmitter and from slot to slot. This scheme is collision-free in the sense that a transmission of an "early" transmitter cannot be interfered with by that of a later one. Consequently, its throughput is monotonically non-decreasing with offered load. Therefore, we will assume that all t transmitters wish to transmit in every slot. Note that a transmitter may unknowingly transmit a message addressed to a blocked destination, since only receivers engaged in the reception of a message addressed to them emit a busy tone.

Since a transmitter does not base its decision to actually transmit on the destination of its message, the most optimistic assumption one can make is that every receiver hears exactly one transmission, and that this transmission is intended for it with probability $1/d_T$. Expected throughput is thus bounded from above by r/d_T.

Casting this result in the context of the SHI's among stations, $r = N_D \cdot c_R$ and $d_T = N_D/c_T$. Consequently, $S_{BTMA} \leq c_T \cdot c_R$.

Proposition 1. *The capacity of any SHI connecting source stations, each with c_T transmitters, to destination stations, each with c_R receivers, with busy-tone multiple-access (BTMA) is at most $c_T \cdot c_R$.* □

The tightness of this upper bound can easily be proved using bus-oriented SHI's, in which $c_T \cdot c_R$ independent subnetworks are constructed, each with a throughput of one for idealized BTMA. We conjecture that this is also the capacity of any non-bus-oriented single-path SHI with idealized BTMA.

Although the performance of BTMA under heavy load is clearly disappointing, it is useful at low loads and skewed traffic pattern. Unlike ALOHA, it is stable and throughput is monotonically non-decreasing with offered load, but one must decide whether the added complexity is warranted.

4. Schedule folding

4.1 Folded schedules

In the basic schedules that attain the high capacity, each schedule slot is allocated to a set of (SS,DS) pairs which may communicate concurrently in that slot, and are all guaranteed to succeed. The idea in a "folded" schedule is that each slot would be allocated to a set of equisized groups of (SS,DS) pairs such that if at most one (arbitrary) member of each group attempts to use the slot, all will succeed. High capacity is retained by selecting the groups such that there is no intergroup interference. Contention among group members for any given slot in which that group is allowed to communicate may be carried out in a variety of ways, including ALOHA. The heavy-load, uniform traffic throughput would thus be equal to that with the unfolded schedule, multiplied by the "utilization factor" of the access scheme used by group members to share a slot.

Definition. *Shortest folded schedule.* Let $f^*(c_T, c_R; N_S, N_D)$ be the minimum number of different time slots for which there exist a compatible wiring W and transmission schedule X such that every (SS, DS) pair is permitted to communicate in some slot (albeit with no guarantee of success), and the heavy-load, uniform-traffic capacity of the interconnection is equal to the maximum capacity with any unfolded schedule to within a constant factor.

Definition. *Folding factor.* We refer to $f()/f^*()$ as the *folding factor*, and note that it is equal to the aforementioned group size.

4.2 The folding factor of a given interconnection

Consider an "expanded" wiring matrix of size $(c_T N_S \times c_R N_D)$. Each of its entries is a single bit, denoting whether a given transmitter can be heard by a given receiver. Partition the matrix into groups of (not necessarily contiguous) identical rows. Based on symmetries in the design of the SHI's under consideration, we can assume that the groups of identical rows are all of equal sizes. Next, repeat the process for columns (the same symmetries apply).

Lemma 2. *The maximum folding factor for a given interconnection is equal to the product of the sizes of the row-groups and column-groups in the wiring matrix.*

Proof. Identical rows represent transmitters that are heard by identical sets of receivers. Thus, if in a given slot one of these transmitters is allowed to transmit to a given destination, letting a different one in the group transmit instead to the same destination would have the same effect (success, blocking, etc.). Similarly, identical columns represent receivers that hear identical sets of transmitters. Therefore, if a transmission intended for one of them is successful, it would be successful if it were intended for any one of the others. So, given a specific (transmitter, receiver) pair that are guaranteed to succeed in a given slot, the transmitter can be replaced with any other transmitter in its group, and the receiver can be replaced with any receiver in its group. The number of (source,destination) pairs that can communicate in this slot (though not concurrently) is therefore the product of the transmitter and receiver group sizes times the concurrency of the interconnection, and the folding factor is simply the product of the group sizes. □

Example. Consider the $(2,1;k,2^k)$ SHI described earlier. From its construction (see Fig. 2) it is clear that no two transmitters are heard by identical sets of receivers, and no two receivers hear identical sets of transmitters. Thus, the folding factor is one, and the minimum schedule length is $k \cdot 2^k/k = 2^k$. In the $(2,1;2^k,2^k)$ SHI constructed from it by adding "layers" of SS's, transmitter-group sizes are all $2^k/k$; the length of the folded schedule is $2^k \cdot 2^k/(k \cdot 2^k/k) = 2^k$, which is equal to the number of stations. (Same as the length of a conventional TDMA round!)

4.3 Folding factor for composite SHI's

Given a wiring matrix, one can easily discover the maximum folding factor. Nevertheless, we proceed to derive expressions for it in order to gain more insight.

Large, high-concurrency SHI's can be constructed by combining smaller ones to form "product" interconnections [6]. We therefore next derive the folding factor of a product interconnection as a function of the folding factors of the constituent interconnections. (The folding factor for SHI's with a single transmitter per station or a single receiver per station is either one or can be trivially derived, as in the foregoing example.) We begin by briefly describing the method by which interconnections can be composed to form larger ones in a manner that retains the high concurrency properties [6].

Consider two interconnections, indexed by $i \in \{1,2\}$. Let S_i, D_i, T_i, R_i satisfy $|S_i| = N_{S_i}$, $|D_i| = N_{D_i}$, $|T_i| = c_{T_i}$, and $|R_i| = c_{R_i}$, where, as before, T_i indexes the transmitters of each station in S_i and R_i indexes the receivers of each station in D_i. Let $W^i = (W_1^i, W_2^i)$ and X^i be compatible wiring and scheduling matrices for the communication between S_i and D_i, such that $W_1^i(s_i,d_i) \in T_i$ and $W_2^i(s_i,d_i) \in R_i$ for $(s_i,d_i) \in S_i \times D_i$.
Let $S = S_1 \times S_2$, $D = D_1 \times D_2$, $T = T_1 \times T_2$ and $R = R_1 \times R_2$. We construct wiring and scheduling matrices for communication between S and D, where T indexes the transmitters of each station in S, and R indexes the receivers of each station in D, as follows. For a pair of stations $s = (s_1,s_2) \in S_1 \times S_2$, $d = (d_1,d_2) \in D_1 \times D_2$, we let $W(s,d) = (W_1(s,d), W_2(s,d)) = (\ (W_1^1(s_1,d_1)W_1^2(s_2,d_2))\ ,\ (W_2^1(s_1,d_1)W_2^2(s_2,d_2))\ )$ and $X(s,d) = (X^1(s_1,d_1)X^2(s_2,d_2))$, where X^1X^2 denotes concatenation of the two strings to form a larger one or a higher-dimensional vector; similarly for $W_i^1W_i^2$.

Lemma 6 in [6] states that the foregoing schedule and wiring are compatible. Also, the length of the round for the composite interconnection is equal to the product of the lengths of the rounds of the constituent interconnections.

Theorem 3. *The maximum folding factor for a composite interconnection is equal to the product of the maximum folding factors for the constituent interconnections.*

Proof.
Lower bound. In Lemma 6 of [6] there is no restriction on the constituent schedules, other than the need for them to be compatible with their own wiring matrices. It follows immediately that the lemma applies to folded schedules as well. Thus, the length of the composite folded schedule will not exceed the product of the lengths of the individual folded schedules, each of which is in turn equal to the length of the respective unfolded schedule divided by the respective folding factor. So, the maximum composite folding factor is greater than or equal to the product of the individual ones.
Upper bound. In [15], it has been shown that one possible layout of a composite interconnection comprises a "column" containing an appropriate number of one of the

constituent interconnections, followed by a column containing an appropriate number of the other. For simplicity of exposition, we refer to the interconnections in the two columns as #1 and #2 interconnections, respectively. Each input of a #1 interconnection is connected to a single transmitter, each output of a #2 interconnection is connected to a single receiver, and there is exactly one connection between each #1 interconnection and every #2 interconnection. Moreover, each output of a #1 interconnection is connected to a single input of a #2 interconnection, and each input of a #2 interconnection is connected to a single output of a #1 interconnection. (The restrictions are mandated by single-hop connectivity of each of the constituent SHI's as well as the composite one, and the provision of a single path between any two stations.) Since all layouts must be equivalent in terms of end-to-end connectivity and disjointness of paths, anything deduced from this layout is valid for all layouts of a composite interconnection.

Since actual transmitters (receivers) are only connected to #1 (#2) interconnections, we will use input (output) groups to refer to ports of the constituent interconnections to which, in stand-alone operation, one would connect the transmitters (receivers) belonging to the same transmitter (receiver) group.

Let $|IG_1|$ and $|IG_2|$ denote the sizes of input groups of a #1 and #2 interconnection, respectively. For transmitters that can be heard by a given receiver to be in the same transmitter group (of the composite interconnection), they must all reach that receiver through the #2 interconnection to which it is connected. Moreover, they must all do so via the same input group of this #2 interconnection; i.e., through at most $|IG_2|$ of its inputs. Also, since there is exactly one connection between any #1 and #2 interconnection, transmitters in the same transmitter group (of the composite interconnection) which are connected to the same #1 interconnection must all be connected to a single input group of that interconnection (otherwise they will reach different #2 interconnections and consequently different receivers).

So, each of the $|IG_2|$ inputs of the #2 interconnection can be reached by at most $|IG_1|$ transmitters in a common transmitter group. Consequently, the total number of transmitters in a transmitter group of the composite interconnection is at most $|IG_1| \cdot |IG_2|$. A similar argument applies to receivers. Combined with Lemma 2, it follows that the folding factor cannot exceed the product of those of the constituent interconnections. □

4.4 Concurrency – folding-factor trade-off

In this section, we use a simple example to show a non-trivial tradeoff between capacity and folding factor in composing interconnections. Let us use the following basic building blocks:

- $(2,1;k,2^k)$: concurrency k; no folding possible; $f = 2^k$.
- single bus with an arbitrary number of stations: concurrency one, $f^* = 1$.

Consider the following two compositions, both producing $(2,2;N,N)$ SHI's:

Composition 1. $(2,1;k,2^k)\ X\ (1,2;2^k,k) \rightarrow (2,2;k2^k,k2^k)$.

- $f() = 2^{2k}$; $f^*() = 2^{2k} > N^2/(\log_2 N)^2$ (close for large N)
- concurrency: k^2 (approximately $\log_2^2 N$); $f()/f^*() = 1$.

Composition 2. This is a two-step construction:
Step 1. $(2,1;k,2^k)\ X\ (1,1;2^k/k,1) \rightarrow (2,1;2^k,2^k)$.
$f() = 2^k \cdot 2^k/k = 2^{2k}/k$; $f^*() = 2^k \cdot 1 = 2^k$. Concurrency: k.
Step 2. $(2,1;2^k,2^k)\ X\ (1,2;2^k,2^k) \rightarrow (2,2;2^{2k},2^{2k})$.

- $f() = 2^{4k}/k^2$; $f^*() = 2^{2k}\ (= N)$
- concurrency: $k^2\ (= \log_2^2 N/4)$; $f()/f^*() = 2^{2k}/k^2 = 4N/\log_2^2 N$.

For equal values of N, the second composition thus yields a much shorter folded schedule, at a penalty of at most a factor of four in concurrency. From this example, Lemma 6 in [6] and Theorem 3, it thus follows that high-capacity SHI's can be composed whose folded schedule lengths are equal to the number of stations.

5. Conclusions

This paper explored schemes for operating high-concurrency, non-bus-oriented SHI's: BTMA and "folded" round-robin schedules.
An idealized form of BTMA gives all receivers an equal chance in every slot, but a small one, to successfully receive a message, and fails to take advantage of the potential of non-bus-oriented SHI's for high concurrency. Deterministic scheduling, in contrast, causes collisions at most receivers but is able to guarantee success to many more than would succeed on average with BTMA.

Schedule "folding" and hierarchical operation (concurrent, non-interfering permission rights to groups of (source,destination) pairs, with contention within each group) were shown to dramatically reduce round length and thus low-load delay, while retaining the high concurrency. This also reduces sensitivity to skews in the traffic pattern.

Further issues to be explored include the impact of folding on delay at various loads and on the sensitivity of throughput to traffic patterns, and dynamic scheduling (picking the schedule slot to be executed in the next time slot), particularly with non-uniform traffic patterns. Finally, an investigation of an unrealistic BTMA, in which receivers emit a different busy tone when blocked though not receiving and a transmitter can tell which receivers are emitting such a tone, may further improve our understanding of what is necessary for attaining a high degree of concurrency.

References

[1] M.A. Marsan and D. Roffinella, "Multichannel local area network protocols", *IEEE JSAC*, vol. 1, no. 5, pp. 885-897, Nov. 1983.

[2] T. Lang, M. Valero and M.A. Fiol, "Reduction of Connections for Multibus Organization," *IEEE Trans. Comp.*, vol. C-32, no. 8, pp. 707-715, Aug. 1983.

[3] Y. Birk, Concurrent Communication among Multi-Transceiver Stations via Shared Media, Ph.D Dissertation, Electrical Engr. Dept., Stanford Univ., Dec. 1986. Also available as tech. rep. CSL-TR-87-321, Mar. 1987.

[4] Y. Birk, F.A. Tobagi and M.E. Marhic, "Bus-oriented interconnection topologies for single-hop communication among multi-transceiver stations", *Proc. IEEE INFOCOM '88*, March 1988.

[5] T.E. Stern, "Linear lightwave network," CTR Tech. Rep. No. 184-90-14, Columbia Univ., 1990.

[6] Y. Birk, N. Linial and R. Meshulam, "On the Uniform-Traffic Capacity of Single-Hop Interconnections Employing Shared Directional Multichannels", *IEEE Trans. IT*, vol. 39, no. 1, pp.186–191, Jan. 1993. (Also IBM Res. Rep. RJ 7859 (72519)).

[7] J. Kilian, S. Kipnis and C.E. Leiserson, "The Organization of Permutation Architectures with Bussed Interconnections", Proc. 28th Annual Symp. on Foundations of Comp. Sci., Los Angeles, CA, Oct. 1987.

[8] M.T. Busche, B. Hajek, "On optical interconnection of stations having multiple transmitters and receivers", *Proc. 1990 Int. Symp. on Information Theo. and its Applications (ISITA '90)*, Hawaii, pp. 967–970, Nov. 1990.

[9] R. A. Barry, "Wavelength routing for all-optical networks", Ph.D Dissertation, EECS Dept., Mass. Inst. of Tech., Sep. 1993, pp. 153–160.

[10] G. Pieris and G. Sasaki, "Scheduling transmissions in WDM broadcast and select networks," Proc. 31st Allerton Conference, Sep. 1993 (to appear).

[11] C. Skawratananond, "Design methodology for bus-oriented networks with single-hop constraint," M.S. Thesis Dept. of Electrical Engr., U. of Texas at Austin, Aug. 1992.

[12] Y. Birk, "Fiber-Optic Bus-Oriented Single-Hop Interconnections among Multi-Transceiver Stations," *IEEE JLT*, vol. 9, no. 12, pp.1657–1664, Dec. 1991.

[13] Y. Birk, "Efficient layout of a passive, single-hop fiber-optic interconnection (among N stations) with capacity $\log_2 N$", in Proc. 29th Annual Allerton Conf. on Commun., Control and Computing, Oct. 1991, Monticello, IL, pp. 322–331.

[14] Y. Birk, "Power-Efficient Layout of a fiber-Optic Multistar that Permits $\log_2 N$ Concurrent Baseband Transmissions among N Stations," *IEEE JLT*, Special issue on broadband optical networks, 1993 (to appear).

[15] Y. Birk, "Power-optimal layout of passive, single-hop, fiber-optic interconnections whose capacity increases with the number of stations", Proc. IEEE INFOCOM '93.

[16] F.A Tobagi and L. Kleinrock, "Packet Switching in Radio Channels: Part II - The Hidden Terminal Problem in Carrier Sense Multiple-Access and the Busy Tone Solution." *IEEE Trans. Commun.* vol. 23, no. 12, pp.1417–1433, Dec. 1975.

Scheduling Transmissions in WDM Broadcast-and-Select Networks[1]

Gerard R. Pieris

Department of Electrical and Computer Engineering
The University of Texas at Austin
Austin, TX 78712-1084
grp@cyndy.ece.utexas.edu

Galen H. Sasaki

Department of Electrical Engineering
University of Hawaii at Manoa
2540 Dole St., Honolulu, HI 96822
sasaki@spectra.eng.hawaii.edu

Abstract. In this paper we consider a broadcast-and-select, wavelength division multiplexed (WDM), optical communication network that is packet switched and time slotted. It is assumed that amount of time for transmitters and receivers to change wavelength is significant. This paper explores how this *tuning duration* can affect the performance of the network. In particular, we present two *all-to-all packet transmission schedules*, that are required to transmit one packet between each input-output pair of the network. The lengths of both schedules are approximately a function of the square root of the tuning duration. The first schedule has near minimum length under the constraint that transmissions are not allowed while transmitters or receivers are changing wavelength. The second has near minimum length under the constraint that the receivers are fixed tuned.

1. Introduction

In this paper we consider the following broadcast-and-select, wavelength division multiplexed (WDM), optical communication network. The physical architecture of the network (as shown in Figure 1) has N inputs, connected to an optical broadcast star,

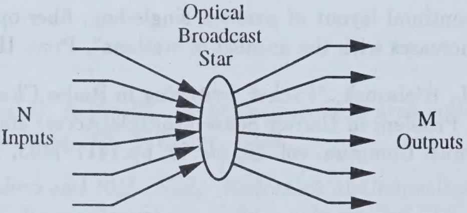


Figure 1. An optical star network with N inputs and M outputs.

---

[1] This research was supported by the National Science Foundation under grant NCR-8958556, the Texas Advanced Research Program under project number 003658-129, and research funds from TRW.

which in turn is connected to M outputs. Each input has t optical transmitters and each output has r optical receivers. It is assumed that the transmitters and receivers are tuned to any one of a set of wavelengths, and the set is large and essentially infinite. Transmitters and receivers that can change wavelengths are called *tunable*, while ones that cannot are called *fixed tuned.* It is assumed that the transmitters are tunable. The network is packet switched, time slotted, and has fixed length packets that have transmission times equal to a time slot. Packets are transmitted within slot boundaries. The time for a transmitter or receiver to change wavelength is called the *tuning duration* and is assumed to be $T > 0$ slots. Note that significant T is expected if packet sizes are small (e.g., ATM cells) and tuning times are relative large (e.g., sometimes in the milliseconds for transmitters and receivers with wide tuning ranges).

The purpose of this paper is to provide insight into the relationship between T and packet delay. We consider the problem of finding a schedule of packet transmissions (and transmitter/receiver tunings) for the network so that one packet is transmitted between each input-output pair. This type of schedule will be called an *all-to-all transmission schedule.* Note that the schedule is for a "uniform" traffic pattern. A measure of packet-delay for the schedule is its *length*, which is the number of time slots it requires. In Sections 2 and 3, we provide upper and lower bound formulas for the minimum length of all-to-all schedules. The formulas relate T with schedule length.

In Section 2, upper bounds on the minimum length of all-to-all transmission schedules are given. Suppose $N = M$, $t = r = 1$, $\sqrt{T}$ is integer, and N is large. Then an upper bound on the length of a minimum length all-to-all transmission schedule is approximately $N(\sqrt{T} + 1)$. An upper bound on the length of a minimum length all-to-all schedule that satisfies the following *tune-transmit separability constraint* is approximately $2N\sqrt{T}$: a schedule satisfying the *tune-transmit separability constraint* is one that does not allow a packet to transmit while a transmitter or receiver is changing wavelength. This constraint is satisfied if a scheduling algorithm is constructed by applying ones designed for satellite-switched TDM systems [7]. (See [8] for example schedules designed for satellite-switched TDM systems.) Tuning transmitters/receivers in an optical network would correspond to reconfiguring an on-board satellite switch. During reconfiguration of a satellite switch, packets cannot transmit.

In Section 3, lower bounds on the length of all-to-all transmission schedules are given. Suppose $N = M$, $t = r = 1$, $\sqrt{T}$ is integer, and $\frac{N}{\sqrt{T}}$ is large. If the receivers are fixed tuned then a lower bound is approximately $N\sqrt{T}$. However, if the receivers are tunable but the schedule satisfies the tune-transmit separability constraint then a lower bound is approximately $2N\sqrt{T}$.

Final remarks are given in Section 4.

To conclude this section, we note that Aggarwal, et al [1] have recently shown that if $N = M$, $t = r = 1$ then all-to-all transmission schedules must have length at least $\frac{N\sqrt{T}}{2\sqrt{3}}$, even if the receivers are tunable. They also report an improved all-to-all transmission schedule that has length at most $N(\sqrt{T} + 0.5)$.

2. Minimum Length Schedules

The proof in the subsequent Theorem 2.1 describe two all-to-all transmission schedules under the assumption $t = r = 1$. One of them satisfies the tune-transmit separability constraint. The theorem is generalized in Theorem 2.2 to arbitrary t and r.

Theorem 2.1. Suppose $t = r = 1$ and the receivers are fixed tuned. Let

$$\begin{aligned} D(k) &:= T + \lceil \tfrac{N}{k} \rceil \lceil \tfrac{M}{k} \rceil + (k-1)\max\{\lceil \tfrac{N}{k} \rceil \lceil \tfrac{M}{k} \rceil, T + \lceil \tfrac{M}{k} \rceil\} && (1) \\ C(k) &:= k(\lceil \tfrac{N}{k} \rceil \lceil \tfrac{M}{k} \rceil + T). && (2) \end{aligned}$$

(a) There is an all-to-all transmission schedule that has length at most $\min_{1 \le k \le \min\{N,M\}} D(k)$.

(b) There is an all-to-all transmission schedule that satisfies the tune-transmit separability constraint and has length at most $\min_{1 \le k \le \min\{N,M\}} C(k)$.

Remarks. Consider the case when $N = M$, $k = \frac{N}{\sqrt{T}}$, and $\sqrt{T}$ is integer. Then $D(k) = N(\sqrt{T}+1) + T - \sqrt{T}$ and $C(k) = 2N\sqrt{T}$. Hence, $\min_k D(k) \le N\left(\sqrt{T}+1\right) + T - \sqrt{T}$ and $\min_k C(k) \le 2N\sqrt{T}$.

Proof. Let k be an arbitrary integer that satisfies $1 \le k \le \min\{N, M\}$. We proceed to describe two schedules. For both schedules, the transmitters (resp., receivers) are grouped into k groups as evenly as possible so that the size of each group is at most $\lceil \frac{N}{k} \rceil$ (resp., $\lceil \frac{M}{k} \rceil$). Denote the transmitter (resp., receiver) groups by $X_0, X_1, ..., X_{k-1}$ (resp., $Y_0, Y_1, ..., Y_{k-1}$). Receiver group Y_i is assigned a unique wavelength w_i and its members are fixed tuned to w_i.

(Proof of Part (b).) We will first describe a schedule that satisfies the tune-transmit separability constraint and has length $C(k)$. The time slots of the schedule are partitioned into subsets of contiguous time slots called *t-frames* and *d-frames* (for transmit and delay frames) of length $\lceil \frac{N}{k} \rceil \lceil \frac{M}{k} \rceil$ and T, respectively. The partition is such that it forms an alternating sequence of t-frames and d-frames, starting with a d-frame and ending with a t-frame. There are k t-frames (and d-frames), and so the length of the schedule is $C(k) = k(\lceil \frac{N}{k} \rceil \lceil \frac{M}{k} \rceil + T)$. The t-frames (and d-frames) are numbered $\{0, 1, ..., k-1\}$. The transmission schedule for X_i is as follows. During d-frame j, the members of X_i change their wavelengths to $w_{(i+j)\bmod k}$. During t-frame j, the members of X_i transmit to members of $Y_{(i+j)\bmod k}$, which is possible because there are at most $\lceil \frac{N}{k} \rceil \lceil \frac{M}{k} \rceil$ transmitter-receiver pairs between a transmitter group and receiver group.

Note that the transmission schedule is all-to-all because each transmitter group transmits to each receiver group. Figure 2 shows an example of a transmission schedule when $N = M = T = 4$ and $k = 2$.

(Proof of Part (a).) We now describe a schedule that has length $D(k)$. The time slots of the schedule are organized as follows. The first T slots is called the *initial*

	Transmitter	Schedule of transmissions and tunings									
X_0	a_1	Tune to w_0	b_1	b_2			Tune to w_1	b_3	b_4		
	a_2	Tune to w_0			b_1	b_2	Tune to w_1			b_3	b_4
X_1	a_3	Tune to w_1	b_3	b_4			Tune to w_0	b_1	b_2		
	a_4	Tune to w_1			b_3	b_4	Tune to w_0			b_1	b_2
		d-frame	t-frame				d-frame	t-frame			

Figure 2. The table shows an all-to-all transmission schedule satisfying the tune-transmt separability constraint when t = r = 1, N = M = T = 4, and k = 2. The transmitters are denoted by $\{a_1, a_2, a_3, a_4\}$ and the receivers are denoted by $\{b_1, b_2, b_3, b_4\}$. The transmitters and receivers are partitioned so that $X_0 = \{a_1, a_2\}$, $X_1 = \{a_3, a_4\}$, $Y_0 = \{b_1, b_2\}$, and $Y_1 = \{b_3, b_4\}$.

tuning phase. The remaining slots form an alternating sequence of t-frames and d-frames, starting and ending with t-frames, and where a d-frame is now composed of $\max\{0, T - (\lceil \frac{N}{k} \rceil - 1)\lceil \frac{M}{k} \rceil\}$ slots. There are k t-frames, and so the length of the schedule is $T + k\lceil \frac{N}{k} \rceil \lceil \frac{M}{k} \rceil + (k-1)\max\{0, T - (\lceil \frac{N}{k} \rceil - 1)\lceil \frac{M}{k} \rceil\}$, which is equal to $D(k)$.

The schedule of transmissions followed by X_i is as follows. Let $\tau_1, \tau_2, ..., \tau_n$ be the transmitters in X_i. Assume, without loss of generality, that $n = \lceil \frac{N}{k} \rceil$, the maximum size of a transmitter group. During the initial tuning phase, X_i tunes to w_i. The t-frames are numbered $\{0, 1, ..., k-1\}$. During t-frame j, X_i transmits to all receivers in $Y_{(i+j)\bmod k}$ as follows: τ_1 transmits in the first $\lceil \frac{M}{k} \rceil$ slots, τ_2 transmits in the second $\lceil \frac{M}{k} \rceil$ slots, and so on. In addition, as soon as a transmitter completes transmitting in t-frame j (for $j < k-1$), it immediately begins changing wavelength to $w_{(i+j+1)\bmod k}$. Note that after transmitting in a t-frame (say t-frame j), a transmitter (say τ_g) has at least T time slots to change wavelengths before transmitting in the next t-frame. More precisely, the number of slots during which τ_g changes wavelength is exactly $(\lceil \frac{N}{k} \rceil - 1)\lceil \frac{M}{k} \rceil + \max\{0, T - (\lceil \frac{N}{k} \rceil - 1)\lceil \frac{M}{k} \rceil\}$, which is at least T. These slots are made up of the $(\lceil \frac{N}{k} \rceil - g)\lceil \frac{M}{k} \rceil$ slots when $\tau_{g+1}, \tau_{g+2}, ..., \tau_n$ transmit in t-frame j, the $\max\{0, T - (\lceil \frac{N}{k} \rceil - 1)\lceil \frac{M}{k} \rceil\}$ slots of a delay frame, and the $(g-1)\lceil \frac{M}{k} \rceil$ slots when $\tau_1, ..., \tau_{g-1}$ transmit in t-frame $j+1$.

The schedule is all-to-all because each transmitter group transmits to each receiver group. Figure 3 shown an example transmission schedule when $N = M = T = 4$ and $k = 2$. ■

Next we discuss generalizing Theorem 2.1 to arbitrary t and r by using the following structures called *pseudo networks*. To form *pseudo networks*, first partition the inputs (resp., outputs) of a network into r (resp., t) subsets, denoted by $\{I_1, I_2, ..., I_r\}$ (resp., $\{O_1, O_2, ... O_t\}$), as evenly as possible. Second, at each input (resp., output) assign unique numbers $\{1, 2, ..., t\}$ (resp., $\{1, 2, ..., r\}$) to the transmitters (resp., receivers). Third, partition all the transmitters and receivers into $t \cdot r$ groups labeled $\{A_{i,j} : 1 \le i \le t, 1 \le j \le r\}$, where $A_{i,j}$ consists of transmitters numbered i in I_j and receivers numbered j in O_i. (Such type of partitions have been used in [3][5] in the context of *bus-oriented*

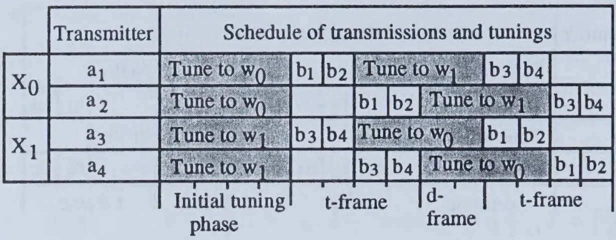


Figure 3. The table shows an all-to-all transmission schedule when t = r = 1, N = M = T = 4, and k = 2. The transmitters are denoted by $\{a_1, a_2, a_3, a_4\}$ and the receivers are denoted by $\{b_1, b_2, b_3, b_4\}$. The transmitters and receivers are partitioned so that $X_0 = \{a_1, a_2\}$, $X_1 = \{a_3, a_4\}$, $Y_0 = \{b_1, b_2\}$, and $Y_1 = \{b_3, b_4\}$.

interconnections.) Figure 4 shows the $A_{i,j}$ when $N = M = 6$, $t = 3$, and $r = 2$. Finally, each $A_{i,j}$ is assigned a large and essentially infinite number of wavelengths that are not shared by other groups. $A_{i,j}$ and its wavelengths is called a *pseudo network* which we denote by $\pi_{i,j}$.

Theorem 2.2. Suppose the receivers are fixed tuned. Let n and m denote $\lceil N/r \rceil$ and $\lceil M/t \rceil$, respectively. Let

$$\overline{D}(k) \quad := \quad T + \lceil \tfrac{n}{k} \rceil \lceil \tfrac{m}{k} \rceil + (k-1)\max\{\lceil \tfrac{n}{k} \rceil \lceil \tfrac{m}{k} \rceil, T + \lceil \tfrac{m}{k} \rceil\} \tag{3}$$

$$\overline{C}(k) \quad := \quad k(\lceil \tfrac{n}{k} \rceil \lceil \tfrac{m}{k} \rceil + T). \tag{4}$$

There is an all-to-all transmission schedule that has length at most $\min_{1 \le k \le \min\{n,m\}} \overline{D}(k)$, and there is an all-to-all transmission schedule that satisfies the tune-transmit separability constraint and has length at most $\min_{1 \le k \le \min\{n,m\}} \overline{C}(k)$.

Proof. Let k be an integer that satisfies $1 \le k \le \min\{n, m\}$. We will describe an all-to-all transmission schedule S_D that that has length $\overline{D}(k)$, and then describe an all-to-all transmission schedule S_C that has length $\overline{C}(k)$ and satisfies the tune-transmit separability constraint. For both schedules, a set of pseudo networks $\{\pi_{i,j} : 1 \le i \le t, 1 \le j \le r\}$ is used. It is assumed, without loss of generality, that there are n transmitters and m receivers in each pseudo network. (Otherwise, we can introduce "dummy" transmitters and receivers to make the assumption true.)

To construct S_D, for each pseudo network, an all-to-all transmission schedule is formed, as in the proof of Theorem 2.1(a). This is possible because within a pseudo network, transmitters and receivers come from different inputs and outputs, respectively. Schedule S_D is the schedules of the pseudo networks run in parallel.

Note that S_D avoids transmission conflicts because pseudo networks have disjoint sets of wavelengths and each pseudo network has a schedule that avoids conflicts within it. Also note that S_D is an all-to-all schedule because each input-output pair has a transmitter and receiver in a common pseudo network (e.g., the inputs of I_j and the outputs of O_i can transmit through $\pi_{i,j}$). Finally, note that the length of S_D is the

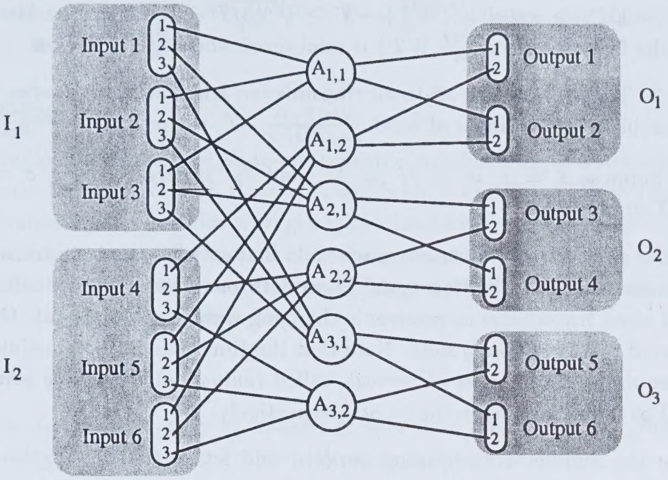


Figure 4. The figure shows the assignment of transmitters and receivers to $A_{i,j}$, for the case N = M = 6, t = 3, and r = 2.

length of a schedule of a pseudo network. Theorem 2.1(a) implies that the length is $\overline{D}(k)$.

Schedule S_C is constructed in the same way as S_D except that the schedule used to prove Theorem 2.1(b) is used rather than the one used to prove Theorem 2.1(a). Using the same arguments as we did for S_D, it can be shown that S_C is an all-to-all schedule with length $\overline{C}(k)$. Finally, note that since pseudo networks have the same number of transmitters and receivers, d-frames of different pseudo networks occur in the same slots. Hence, S_C satisfies the tune-transmit separability constraint. ■

3. Lower Bounds On Schedule Lengths

In this section, we present two lower bounds on the length of all-to-all transmission schedules.

Theorem 3.1. Suppose the network has fixed tuned receivers and $t = r = 1$. Then an all-to-all transmission schedule has length at least $\sqrt{NMT} - T$.

Proof. Assume that the receivers are tuned to W wavelengths. Since at most W wavelengths may be used at any time and there are $N \cdot M$ packets to transmit, an all-to-all transmission schedule requires at least $\frac{N \cdot M}{W}$ slots. In addition, an all-to-all transmission schedule requires that each input tune its transmitter to each of the W wavelengths so that it can transmit a packet to each output. Therefore, an all-to-all transmission schedule requires at least $(W-1)T$ time slots.

Hence, the time required for an all-to-all transmission schedule is at least

$\max\{\frac{NM}{W}, (W-1)T\} \geq \max\{\frac{NM}{W}, WT\} - T \geq \sqrt{NMT} - T$, where the last inequality is due to the fact that $\max\{\frac{NM}{W}, WT\}$ is minimized when $\frac{NM}{W} = WT$. ■

Theorem 3.2. The length of an all-to-all transmission schedule that satisfies the tune-transmit separability constraint is at least $\frac{NM(T+\sigma)}{\sigma+\sqrt{NtMr\sigma}} - T$, where $\sigma := \frac{(T+\sqrt{T^2+NtMrT})^2}{NtMr}$.

Remarks. Suppose $t = r = 1$, $N = M$, and $\frac{N^2}{T}$ is large. Then $\sigma \approx T$ and $\frac{NM(T+\sigma)}{\sigma+\sqrt{NtMr\sigma}} - T \approx 2N\sqrt{T}$.

Proof. Consider an all-to-all transmission schedule that satisfies the tune-transmit separability constraint, and let L be its length. Time slots for the schedule are referred to as *tuning* slots if some transmitter or receiver is changing wavelength during it. Other time slots are referred to as *nontuning* slots. Note that the time slots of the schedule form an alternating sequence of two types of *periods*, called *tuning* and *nontuning periods*, that are composed of tuning and nontuning slots, respectively.

Let k be the number of nontuning periods, and let p_j be the length of the j^{th} nontuning period. For $2 \leq j \leq k$, let q_j be the length of the tuning period immediately before the j^{th} nontuning period. Hence, $L \geq p_1 + \sum_{j=2}^{k}(q_j + p_j)$. Let q_1 be defined to be equal to T. We will proceed to show that $\sum_{j=1}^{k}(q_j + p_j) \geq \frac{NM(T+\sigma)}{\sigma+\sqrt{NtMr\sigma}}$, which implies $L \geq \frac{NM(T+\sigma)}{\sigma+\sqrt{NtMr\sigma}} - T$ and the theorem.

Let u_j be the number of packets transmitted during the j^{th} nontuning period. The total number of packets transmitted by the schedule is $\sum_{j=1}^{k} u_j$ because, by the tune-transmit separability constraint, packets are transmitted only during nontuning periods. Then $\sum_{j=1}^{k} u_j = NM$ because the schedule is all-to-all. In addition, if $\gamma := \max_{1 \leq j \leq k} \frac{u_j}{q_j+p_j}$ then $\gamma \sum_{j=1}^{k}(q_j + p_j) \geq \sum_{j=1}^{k} u_j = NM$. Thus, $\sum_{j=1}^{k}(q_j + p_j) \geq \frac{NM}{\gamma}$. To complete the proof we will show that $\gamma \leq \frac{\sigma+\sqrt{NtMr\sigma}}{T+\sigma}$.

Consider the j^{th} nontuning period. Let $w_1, w_2, ..., w_c$ be the wavelengths used during the period. Let x_i and y_i be the number of transmitters and receivers, respectively, tuned to w_i. The number of packets transmitted over w_i is at most $x_i \cdot y_i$ because at most one packet is transmitted between an input-output pair. Also, the number is at most p_i because at most one packet is transmitted over w_i per slot. Hence, $u_j \leq \sum_{i=1}^{c} \min\{x_i y_i, p_j\}$. If $\{x_i, y_i\}$ are treated as real-valued variables then

$$u_i \leq \max\left\{\sum_{i=1}^{c} \min\{x_i y_i, p_i\} : \sum_{i=1}^{c} x_i \leq Nt, \sum_{i=1}^{c} y_i \leq Mr, \text{ and for } 1 \leq i \leq c,\ x_i, y_i \geq 0\right\} \tag{5}$$

There is an optimal solution for the right hand side of (5) such that $x_i y_i \leq p_i$ for $1 \leq i \leq c$. Otherwise, for each $x_i y_i > p_i$, x_i can be decreased until $x_i y_i = p_i$, and without changing the value of $\sum_{i=1}^{c} \min\{x_i y_i, p_i\}$ or violating constraints. Therefore,

$$u_i \leq \max\left\{\sum_{i=1}^{c} x_i y_i : \sum_{i=1}^{c} x_i \leq Nt, \sum_{i=1}^{c} y_i \leq Mr, \text{ and for } 1 \leq i \leq c,\ x_i y_i \leq p_i, x_i, y_i \geq 0\right\} \tag{6}$$

The right hand side of (6) is at most $p_i + \sqrt{NtMrp_i}$ by an application of Lemma A in the Appendix. Thus, $\frac{u_i}{q_i+p_i} \le \frac{p_i+\sqrt{NtMrp_i}}{q_i+p_i} \le \frac{p_i+\sqrt{NtMrp_i}}{T+p_i}$, which is maximized when $p_i = \sigma$. Hence, $\gamma \le \frac{\sigma+\sqrt{NtMr\sigma}}{T+\sigma}$. ■

4. Conclusions

We studied the problem of all-to-all transmission scheduling in a time-slotted, optical star network. Suppose $N = M$, $t = r = 1$, and both N and T are large. Then there exist all-to-all transmission schedules of length approximately $N\sqrt{T}$ slots. Hence, the schedule lengths increase as a square root of the tuning duration. If an all-to-all transmission schedule satisfies the tune-transmit separability constraint then its length must be at least twice that of approximately $N\sqrt{T}$.

Appendix

Lemma A. Let n, m, c, and p be integers greater than zero. Then $p + \sqrt{nmp}$ is at least the maximum of the following Problem P_A, with variables $\{x_i, y_i\}_{i=1}^c$:

$$\text{Maximize} \quad \sum_{i=1}^{c} x_i \cdot y_i \tag{7}$$

$$\text{Subject to} \quad \sum_{i=1}^{c} x_i \le n, \ \sum_{i=1}^{c} y_i \le m, \tag{8}$$

$$\text{and, for } 1 \le i \le c, \ \ x_i \cdot y_i \le p \ \text{ and } \ x_i, y_i \ge 0. \tag{9}$$

Proof. First, we will show that there is an optimal solution $\{\overline{x}_i, \overline{y}_i\}_{i=1}^c$ for P_A that satisfies the following Property (P.1): there is an integer h such that

$$\overline{x}_i \cdot \overline{y}_i = \begin{cases} p, & \text{if } i < h \\ 0, & \text{if } i > h \end{cases} \tag{10}$$

Let $\{\overline{x}_i, \overline{y}_i\}_{i=1}^c$ be an optimal solution, and assume, without loss of generality, that $\overline{x}_1 \ge \overline{x}_2 \ge \dots \ge \overline{x}_c$. Since $\{\overline{x}_i, \overline{y}_i\}_{i=1}^c$ is an optimal solution, $\{\overline{y}_i\}_{i=1}^c$ must be an optimal solution for the following linear programming problem:

$$\max\left\{\sum_{i=1}^{c} \overline{x}_i y_i : \ \sum_{i=1}^{c} y_i \le m, \ \text{ and for } \ 1 \le i \le c, \ \ y_i \ge 0, \ \overline{x}_i y_i \ge p\right\} \tag{11}$$

Since the coefficients $\{\overline{x}_i\}$ of the linear programming problem are nonincreasing with i, $\{\overline{y}_i\}_{i=1}^c$ must satisfy (P.1).

Second, we will show that there is an optimal solution $\{\overline{x}_i, \overline{y}_i\}_{i=1}^c$ that satisfies (P.1) and $\overline{x}_1 = \overline{x}_2 = \dots = \overline{x}_{h-1}$. Since the case $h = 1$ is trivial, we assume $h > 1$. Suppose $\{\hat{x}_i, \hat{y}_i\}_{i=1}^c$ is an optimal solution that satisfies (P.1). Define $\{\overline{x}_i, \overline{y}_i\}_{i=1}^c$ so that

$$\overline{x}_i = \begin{cases} \frac{1}{h-1}\sum_{i=1}^{h-1} \hat{x}_i, & \text{if } i < h \\ \hat{x}_i, & \text{if } i \ge h \end{cases} \tag{12}$$

$$\overline{y}_i = \begin{cases} p/\overline{x}_i, & \text{if } i < h \\ \hat{y}_i, & \text{if } i \geq h \end{cases} \tag{13}$$

Note that $\{\overline{x}_i, \overline{y}_i\}_{i=1}^{c}$ satisfies (P.1) and $\overline{x}_1 = \overline{x}_2 = \ldots = \overline{x}_{h-1}$. Also note that, since $\sum_{i=1}^{c} \hat{x}_i \hat{y}_i = \sum_{i=1}^{c} \overline{x}_i \overline{y}_i$, it is an optimal solution if it satisfies the constraints of P_A.

It is trivial to check the inequalities $\overline{x}_i \cdot \overline{y}_i \leq p$, $\overline{x}_i \geq 0$, and $\overline{y}_i \geq 0$ for $i = 1, 2, \ldots, c$. It is also trivial to check that $\sum_{i=1}^{c} \hat{x}_i = \sum_{i=1}^{c} \overline{x}_i$, which implies $\sum_{i=1}^{c} \overline{x}_i \leq n$. To check the inequality $\sum_{i=1}^{c} \overline{y}_i \leq m$, first note that

$$\min\left\{ \sum_{i=1}^{h-1} \frac{p}{x_i} : \quad \sum_{i=1}^{h-1} x_i = \sum_{i=1}^{h-1} \hat{x}_i \text{ and for } 1 \leq i < h,\ x_i \geq 0 \right\} \leq \sum_{i=1}^{h-1} \hat{y}_i \tag{14}$$

because $\hat{y}_i = p/\hat{x}_i$ for $i < h$. The optimization problem on the left hand side of (14) has convex cost and linear constraints. Thus, the Kuhn-Tucker Conditions for Optimality [9] can be applied to verify that $\{\overline{x}_i\}_{i=1}^{h-1}$ minimizes the left hand side of (14). Hence, $\sum_{i=1}^{h-1} \overline{y}_i = \sum_{i=1}^{h-1} p/\overline{x}_i \leq \sum_{i=1}^{h-1} \hat{y}_i$. This implies $\sum_{i=1}^{c} \overline{y}_i \leq \sum_{i=1}^{c} \hat{y}_i \leq m$.

Finally, suppose $\{\overline{x}_i, \overline{y}_i\}_{i=1}^{c}$ is an optimal solution for P_A, that satisfies (P.1) and $\overline{x}_1 = \overline{x}_2 = \ldots = \overline{x}_{h-1}$. Then

$$\sum_{i=1}^{c} \overline{x}_i \overline{y}_i = \sum_{i=1}^{h} \overline{x}_i \overline{y}_i \leq h \cdot p \tag{15}$$

$$(h-1)\overline{x}_1 = \sum_{i=1}^{h-1} \overline{x}_i \leq n \tag{16}$$

$$(h-1)p/\overline{x}_1 = \sum_{i=1}^{h-1} \overline{y}_i \leq m \tag{17}$$

and $\overline{x}_1 > 0$. Hence, the maximum for P_A cannot exceed

$$\max\{h \cdot p : \quad h \text{ and } x \text{ satisfy } (h-1)x \leq n,\ (h-1)p/x \leq m,\ x > 0\} \tag{18}$$

$$= \max\{h \cdot p : \quad h - 1 \leq n/x,\ h - 1 \leq mx/p,\ x > 0\} \tag{19}$$

$$= p \cdot \max_{x>0}(1 + p \cdot \min\{n/x, mx/p\}) \tag{20}$$

$$= p + \sqrt{nmp} \tag{21}$$

where the last equality comes about because the minimum in (20) is achieved when $n/x = xm/p$. ■

References

[1] A. Aggarwal, A. Bar-Noy, D. Coppersmith, R. Ramaswami, B. Schieber, and M. Sudan, "Efficient routing and scheduling algorithms for optical networks," Tech. Rept. RC 18967, IBM Research Report, June 1993.

[2] Krishna Bala, Thomas E. Stern and Kavita Bala, "Algorithms for Routing in a Linear Lightwave Network," *IEEE INFOCOM '91 Proceedings*, Bal Harbour, Florida, pp. 1-9.

[3] Y. Birk, F. A. Tobagi and M. E. Marhic, "Bus-oriented Interconnection Topologies for Single-hop Communication among Multi-transceiver Stations," *IEEE INFOCOM '88 Proceedings*, New Orleans, Louisiana, March 1988, pp. 558-567.

[4] Charles A. Bracket, "Dense Wavelength Division Multiplexing Networks: Principles and Applications," *IEEE Journal on Selected Areas in Communications*, vol. 8, no. 6, Aug. 1990, pp. 948-964.

[5] M. T. Busche, "Throughput, Delay and Power Losses in Bus-oriented Optical Networks with Multiple Transmitters and Receivers per Station," University of Illinois at Urbana-Champaign Report UILU-ENG-91-2205, Jan. 1991.

[6] M. Busche and B. Hajek, "On Optical Interconnection of Stations having Multiple Transmitters and Receivers," *Proceedings of the 1990 International Symposium on Information Theory and Its Applications*, Honolulu, Hawaii, Nov. 1990, pp. 967-970.

[7] A. Ganz and Y. Gao, "A Time-Wavelength Assignment Algorithm for a WDM Star Network," *IEEE INFOCOM '92 Proceedings*, Florence, Italy, May 1992, pp. 2144-2150.

[8] I. Gopal and C.K. Wong, "Minimizing the number of switchings in an SS/TDMA system," *IEEE Trans. on Commun.*, vol. COM-33, No. 6, pp. 497-501, June 1985.

[9] D. Luenberger, "Linear and Nonlinear Programming," Addison-Wesley Publishing Company, Reading M.A., 1973.

A Class of Non-Linear Block Codes over Z_{2^m} for the Lee Metric

Antonio C. F. Pessoa
CPqD/TELEBRÁS
C. P. 1579, Campinas, SP, Brazil 13085-900
e-mail: franca@dragon.cpqd.ansp.br

Dalton S. Arantes[1]
Department of Communications
School of Electrical Engineering, UNICAMP
C. P. 6101, Campinas, SP, Brazil 13081
e-mail: dalton@decom.fee.unicamp.br.

Abstract

A class of non-linear block codes over the ring of integers modulo 2^m, Z_{2^m}, is presented for the Lee metric. These codes are constructed from m binary linear block codes and can be decoded in successive steps, where each binary code is decoded in each step. Therefore, known decoding algorithms for binary codes can be used to decode these multidimensional codes. Theoretical results are presented for distance properties and decoding performance.

1. Introduction

Coding techniques with the Lee metric have been used in many digital communication systems with amplitude and/or phase modulation [01,02,03]. In these systems, the input symbols belonging to a ring of integers are mapped into a constellation of modulated signals with different amplitudes and/or phases.

In general, the Lee distance between two signals is a monotonically decreasing function of the probability of receiving one of these signals given that the other is transmitted. Codes with the Lee metric are efficient for such channels because the most likely channel errors correspond to error patterns with small Lee weights. It can be shown that the Lee metric is optimal for channel noise with exponential distribution and, for high signal-to-noise ratio, it provides good results for the Gaussian noise channel. Moreover, decoders with the Lee metric are normally less complex than decoders designed for the Euclidian metric.

Only a few works on coding under the Lee metric have been published [03,04,05,06]. In these cases, the codes are mainly designed with symbols from Z_p, where p is a prime, and the extension to Z_{p^m}, $m > 1$, is not possible. On the other hand, the majority of works on coding over the ring of integers [08,09,10,11], has been based on the Hamming distance. Unfortunately, for these codes the minimum Lee distance is severely limited to the corresponding minimum Hamming distance. More recently, codes over the ring of integers have also been designed for the Euclidian metric [07].

---

[1] On leave at the Dept. of Elect. and Computer Engineering - UNM, EECE Building - Room 110, Albuquerque, NM 87131, e-mail: arantes@jemez.eece.unm.edu. This work was supported by CPqD-TELEBRAS (Contract 387/90), Fundação de Amparo à Pesquisa do Estado de São Paulo(FAPESP), Conselho Nacional de Desenvolvimento Científico e Tecnológico(CNPq) and The Ibero-American Science and Technology Education Consortium(ISTEC).

Since most practical modulation systems use a number of signal elements that is a power of 2, in the present work the codes are designed over the ring of integers modulo 2^m, where m is an integer.

A number of useful parameters are defined as follows. Let

$$\mathbf{z}^T = [z_0 \, z_1 \cdots z_{n-1}] \tag{1}$$

be a codeword of length n, where z_i belongs to the alphabet

$$Z_{2^m} = \{0, 1, \cdots, 2^m - 1\} \tag{2}$$

The Lee weight of $\mathbf{z}$ is defined as follows

$$\omega_{2^m}(\mathbf{z}) = \sum_{j=0}^{n-1} \Gamma_{2^m}(z_j) \tag{3}$$

where $\Gamma_{2^m}(v)$, the Lee weight of the symbol v, is given by

$$\Gamma_{2^m}(v) = (v) \bmod 2^m \text{ or } (-v) \bmod 2^m \text{ with } 0 \leq \Gamma_{2^m}(v) \leq 2^{m-1} \tag{4}$$

The Lee distance between two vectors is defined as the Lee weight of their difference. It can be shown that the Lee distance is a metric.

Let

$$\mathbf{u}^{(r)^T} = [u_0^{(r)} \, u_1^{(r)} \cdots u_{n-1}^{(r)}], \; u_j^{(r)} \in Z_{2^m}, \; j = 0, 1, \ldots, n-1 \tag{5}$$

be the r-th codeword of a block code C with blocklength n over Z_{2^m}.

The minimum Lee distance of the code C is given by

$$D_L = \min_{r \neq s} \; \omega_{2^m}(\mathbf{u}^{(r)} - \mathbf{u}^{(s)}) \; ; \; \mathbf{u}^{(r)}, \mathbf{u}^{(s)} \in \mathrm{C} \tag{6}$$

It has been proved that if a code C has minimum Lee distance D_L, then all the error patterns of Lee weight less than $D_L/2$ can be corrected [04].

A code C over Z_{2^m} is linear when

$$a\mathbf{u}^{(r)} + b\mathbf{u}^{(s)} \in \mathrm{C} \quad \forall \, \mathbf{u}^{(r)}, \mathbf{u}^{(s)} \in \mathrm{C} \quad \forall \, a, b \in Z_{2^m} \tag{7}$$

The codes introduced in the present work are non-linear, except for the trivial case where $m = 1$, when they coincide with the binary codes. Although the idea of non-linearity is usually associated with high decoding complexity, the present class of non-linear codes is relatively easy to decode, since they are constructed from a set of binary linear codes. The redundancy of a code C over Z_{2^m} is defined as follows [12]

$$R = n - \log_{2^m}(\text{number of codewords in C}) \tag{8}$$

For a code C over Z_{2^m} with blocklength n and minimum Lee distance $D_L = 2t + 1$, the sphere packing bound for R is given by

$$R \geq \frac{1}{m} \log_2 \left[1 + \sum_{i=1}^{t} F_m(i, t) \begin{pmatrix} n \\ i \end{pmatrix} \right] \tag{9}$$

m	$F(1,1)$	$F(1,2)$	$F(2,2)$	$F(1,3)$	$F(2,3)$	$F(3,3)$
2	2	3	4	3	8	8
≥ 3	2	4	4	6	12	8

Table 1: Values of $F(i,t)$ for $t = 1, 2, 3$.

where $F_m(i,t)$ is the number of i-tuples with non-zero elements from Z_{2^m} with Lee weight less than or equal to t.

The values of $F_m(i,t)$ for $t \leq 3$ are presented in Table 1.

2. Code design

The construction of the present class of non-linear codes is accomplished as follows.

Let C_0, C_1, ..., C_{m-1} be a set of m linear binary block codes with parameters (n, k_i, d_i), $i = 0, 1, 2, \ldots, m-1$, where k_i is the dimension and d_i is the minimum Hamming distance of C_i.

A non-linear code C over Z_{2^m}, whose construction is based on the set C_0, C_1, ..., C_{m-1} of linear binary codes, will be defined as the collection of all codewords of the form

$$\mathbf{u}^T = [u_0\, u_1 \cdots u_{n-1}], \quad u_j \in Z_{2^m} \tag{10}$$

where u_j, $j = 0, 1, \ldots, n-1$, is given by

$$u_j = \sum_{i=0}^{m-1} x_{i,j} 2^i \ , \ x_{i,j} \in \{0, 1\} \tag{11}$$

and where

$$\mathbf{x}_i^T = [x_{i,0}\, x_{i,1} \cdots x_{i,n-1}] \tag{12}$$

is a codeword of C_i, for $i = 0, 1, \ldots, m-1$.

The procedure for designing code C is better illustrated as shown in Figure 1.

It is easy to see that the redundancy of a code in this class is given by

$$R = n - \frac{\sum_{i=0}^{m-1} k_i}{m} \tag{13}$$

The following result holds for this class of codes.

Theorem 1 *The minimum Lee distance, D_L, of a code C over Z_{2^m}, whose construction is based on the binary codes C_0, C_1, ..., C_{m-1}, with parameters (n, k_i, d_i), $i = 0, 1, 2, \ldots, m-1$, is given by*

$$D_L = \min_{0 \leq i \leq m-1} d_i 2^i \tag{14}$$

$2^0\times$	$x_{0,0}$	$x_{0,1}$	$\cdots$	$x_{0,n-1}$	$\Leftarrow$ belongs to code C_0 over $\{0,1\}$
	$+$	$+$	$\cdots$	$+$	
$2^1\times$	$x_{1,0}$	$x_{1,1}$	$\cdots$	$x_{1,n-1}$	$\Leftarrow$ belongs to code C_1 over $\{0,1\}$
	$+$	$+$	$\cdots$	$+$	
	$\vdots$	$\vdots$	$\cdots$	$\vdots$	
	$+$	$+$	$\cdots$	$+$	
$2^{m-1}\times$	$x_{m-1,0}$	$x_{m-1,1}$	$\cdots$	$x_{m-1,n-1}$	$\Leftarrow$ belongs to code C_{m-1} over $\{0,1\}$
	$=$	$=$	$\cdots$	$=$	
	u_0	u_1	$\cdots$	u_{n-1}	$\Leftarrow$ belongs to code C over Z_{2^m}

Figure 1: Construction of a code C over Z_{2^m}

3. A decoding algorithm for C

In this section we present a simple algorithm for decoding a code C in the above class. We can show that this decoder can decode any error pattern with Lee weight less than $D_L/2$, for any code with minimum Lee distance equal to D_L.

Let C be a code over Z_{2^m} with blocklength n, whose construction is based on the binary codes $C_0, C_1, \ldots, C_{m-1}$, and let

$$\mathbf{v}^T = [v_0\, v_1\, \cdots\, v_{n-1}],\ v_j \in Z_{2^m} \tag{15}$$

be the received vector.

After m consecutive decoding steps $(0, 1, \ldots, m-1)$, the decoder will produce at its output the codeword

$$\hat{\mathbf{u}}^T = [\hat{u}_0\, \hat{u}_1\, \cdots\, \hat{u}_{n-1}]\ ,\ \hat{u}_j \in Z_{2^m} \tag{16}$$

where

$$\hat{u}_j = \sum_{i=0}^{m-1} \hat{x}_{i,j} 2^i\ ,\ \hat{x}_{i,j} \in \{0,1\} \tag{17}$$

In the q-th decoding step, $0 \leq q \leq m-1$, which is performed after completion of steps $0, 1, \ldots, q-1$, the decoder concentrates exclusively on the code C_q, from which it obtains the binary codeword

$$\hat{\mathbf{x}}_q^T = [\hat{x}_{q,0}\, \hat{x}_{q,1}\, \cdots\, \hat{x}_{q,n-1}]\ ,\ \hat{x}_{q,j} \in \{0,1\} \tag{18}$$

This codeword is necessary for calculating the vector over $Z_{2^{q+1}}$

$$\alpha^{(q)^T} = [\alpha_0^{(q)}\, \alpha_1^{(q)}\, \cdots\, \alpha_{n-1}^{(q)}]\ ,\ \alpha_j^{(q)} \in Z_{2^{q+1}} \tag{19}$$

which represents an approximation to $\hat{\mathbf{u}}$ mod 2^{q+1} such that

$$\alpha_j^{(q)} = \begin{cases} \alpha_j^{(q-1)} + \hat{x}_{q,j} 2^q & \text{for } q = 1, 2, \ldots, m-1 \text{ and } j = 0, 1, \ldots, n-1 \\ \hat{x}_{0,j} & \text{for } q = 0 \text{ and } j = 0, 1, \ldots, n-1 \end{cases} \tag{20}$$

It is easy to verify that

$$\hat{u}_j = \alpha_j^{(m-1)} \; , \; j = 0, 1, \ldots, n-1 \tag{21}$$

In the q-th decoding step the decoder searchs for the codeword $\hat{\mathbf{x}}_q$, belonging to $\mathrm{C}_q = \{\mathbf{c}_r^{(q)}\}$ and given by (18), which achieves

$$\min_{\mathbf{c}_r^{(q)},\gamma_r} \quad \omega_{2^m}(\mathbf{v} - \gamma_r) \tag{22}$$

where

$$\mathbf{c}_r^{(q)} = [c_{r,0}^{(q)} \, c_{r,1}^{(q)} \cdots c_{r,n-1}^{(q)}] \; , \; c_{r,j}^{(q)} \in \{0,1\} \tag{23}$$

is the r-th codeword of C_q and

$$\gamma_r^T = [\gamma_{r,0} \, \gamma_{r,1} \cdots \gamma_{r,n-1}] \; , \; \gamma_{r,j} \in Z_{2^m} \tag{24}$$

depends on $\mathbf{c}_r^{(q)}$ and on $\alpha^{(q-1)}$ according to the expression

$$\gamma_{r,j} \bmod 2^{q+1} = \alpha_j^{(q-1)} + c_{r,j}^{(q)} 2^q \; , \; j = 0, 1, \ldots, n-1 \tag{25}$$

Therefore,

$$\min_{\mathbf{c}_r^{(q)},\gamma_r} \omega_{2^m}(\mathbf{v} - \gamma_r) = \min_{\mathbf{c}_r^{(q)},\gamma_r} \sum_{j=0}^{n-1} \Gamma_{2^m}(v_j - \gamma_{r,j}) = \min_{\mathbf{c}_r^{(q)}} \sum_{j=0}^{n-1} \Gamma_{2^{q+1}}(v_j - \alpha_j^{(q-1)} - c_{r,j}^{(q)} 2^q) \tag{26}$$

It is easy to see that

$$\Gamma_{2^{q+1}}(v_j - \alpha_j^{(q-1)} - c_{r,j}^{(q)} 2^q) = \begin{cases} 2^q(1 - \beta_j^{(q)}) & , \; c_{r,j} = 1 \\ 2^q \beta_j^{(q)} & , \; c_{r,j} = 0 \end{cases} \tag{27}$$

where

$$\beta_j^{(q)} = \frac{\Gamma_{2^{q+1}}(v_j - \alpha_j^{(q-1)})}{2^q} \; , \; j = 0, 1, \ldots, n-1 \tag{28}$$

and

$$[\beta_0^{(q)} \, \beta_1^{(q)} \ldots \beta_{n-1}^{(q)}] = \beta^{(q)} \tag{29}$$

Therefore, the decoding rule given by expression (22) is equivalent to

$$\min_{\mathbf{c}_r^{(q)}} \sum_{j=0}^{n-1} \left| c_{r,j}^{(q)} - \beta_j^{(q)} \right| \tag{30}$$

which coincides with the decoding rule for a binary code C_q in a binary input, (2^q+1)-ary output channel, since $\beta_j^{(q)}$ takes values in the set $\{0, 1/2^q, 2/2^q, 3/2^q, \ldots, 1\}$.

It is important to emphasize that, although the global decoding of C is based on hard-decision symbols, the decoding of each binary code in a given step is accomplished over a finite alphabet of rational numbers. Moreover, there are many algorithms available in the literature which can be used for decoding binary codes over channels with Q-ary output. Examples of such algorithms are the WED (Weighted Erasure Decoding) by Weldon[13], the GMD (Generalized Minimum Distance Decoding) by Forney[14], and others [15,16].

The following result holds for the above decoding algorithm.

Step	v	0	1	7	4	7	7	7	7	5	7	7	0	0	1	0	0	6	0	0	0
0	$\beta^{(0)}$	.0	1.	1.	.0	1.	1.	1.	1.	1.	1.	1.	.0	.0	1.	.0	.0	.0	.0	.0	.0
	$\hat{x}_0$	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
	$\alpha^{(0)}$	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
1	$\beta^{(1)}$	.5	.0	1.	.5	1.	1.	1.	1.	.0	1.	.5	.0	.0	.5	.0	.0	1.	.0	.0	.0
	$\hat{x}_1$	1	0	1	0	1	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0
	$\alpha^{(1)}$	3	1	3	1	3	3	1	3	1	3	0	0	0	0	0	0	0	0	0	0
2	$\beta^{(2)}$	.75	.0	1.	.75	1.	1.	.5	1.	1.	1.	.25	.0	.0	.25	.0	.0	.5	.0	.0	.0
	$\hat{x}_2$	1	0	1	1	1	1	0	1	1	1	0	0	0	0	0	0	0	0	0	0
	$\alpha^{(2)}$	7	1	7	5	7	7	1	7	5	7	0	0	0	0	0	0	0	0	0	0

Table 2: Four examples of the decoding algorithm for code C (Note that $\alpha^{(2)} = \hat{\mathbf{u}}$).

Theorem 2 *Let* C *be a code over* Z_{2^m} *with a minimum Lee distance equal to* D_L, *constructed from* $C_0, C_1, \ldots, C_{m-1}$, *with parameters* (n, k_i, d_i), $i = 0, 1, 2, \ldots, m-1$, *respectively. If the above m-steps decoding algorithm is used, then any error pattern with Lee weight less than* $D_L/2$ *can be corrected.*

Simple examples for the above decoding algorithm are presented bellow.

3.1 Examples

Let C be a code over Z_8, constructed from the binary codes C_0, C_1 and C_2, with parameters (5,1,5), (5,2,3) and (5,4,2), respectively, and with the following generator matrices

$$G_0 = [1\ 1\ 1\ 1\ 1] \quad G_1 = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \end{bmatrix} \quad G_2 = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

The minimum Lee distance of code C is given by

$$D_L = \min(2^0 \times 5,\ 2^1 \times 3,\ 2^2 \times 2) = 5$$

In Table 2 we present the results of the decoding algorithm for 4 different received vectors.

4. A number of codes over Z_4 and Z_8.

In Tables 3, 4 and 5 we present a number of optimized codes over the rings Z_4 and Z_8, with minimum Lee distances $D_L = 3, 5, 7$, respectively. These codes were obtained according to the technique presented in Section 3, where we have used the best known binary linear codes of blocklengths less than or equal to 128, tabulated in [12,17].

In these tables we also present the redundancy R of each code, as well as the corresponding lower bound given by (09). For blocklengths not presented in these tables,

$D_L = 3$ $(d_0 = 3, d_1 = 2, d_2 = 1)$							
n	k_0	k_1	k_2	Codes over $Z_4(C_0, C_1)$		Codes over Z_8 (C_0, C_1, C_2)	
				R	L. Bound(09)	R	L. Bound(09)
3	1	2	3	1.500000	1.403677	1.000000	0.935785
7	4	6	7	2.000000	1.953445	1.333333	1.302297
15	11	14	15	2.500000	2.477098	1.666667	1.651399
31	26	30	31	3.000000	2.988640	2.000000	1.992427
63	57	62	63	3.500000	3.494342	2.333333	2.329562
127	120	126	127	4.000000	3.997177	2.666667	2.664784

Table 3: Codes over Z_4 and Z_8 with $D_L = 3$.

the best codes can be obtained by shortening the binary codes of the corresponding multilevel code with blocklength immediately above the desired one.

Additional results have been obtained for these codes, but they will not be presented here. For example, we have found that they are better than the codes introduced by Nakamura[03]. We have also compared the present codes with the quaternary codes introduced in [18]. In this latter case, the minimum Lee distance is equal to the minimum Hamming distance of the corresponding quaternary code.

5. Conclusions

The present work introduces a class of multilevel codes over Z_{2^m}, optimized with respect to the Lee metric. These codes are in general more efficient than corresponding known codes which have appeared in the literature. We have also presented a relatively simple multistage decoding algorithm for this class of codes.

6. Acknowledgments

It is a pleasure to acknowledge those who have helped us in the preparation of this paper. Special thanks are due to Prof. Ramiro Jordan and the Ibero-American Science and Technology Education Consortium-ISTEC for their support during the final phase of this work.

$D_L = 5$ $(d_0 = 5, d_1 = 3, d_2 = 2)$							
n	k_0	k_1	k_2	Codes over $Z_4(C_0, C_1)$		Codes over Z_8 (C_0, C_1, C_2)	
				R	L. Bound(09)	R	L. Bound(09)
5	1	2	4	3.500000	2.903677	2.666667	1.976912
7	1	4	6	4.500000	3.363960	3.333333	2.273393
8	2	4	7	5.000000	3.549016	3.666667	2.393303
15	7	11	14	6.000000	4.432093	4.333333	2.969964
23	14	18	22	7.000000	5.039742	5.000000	3.369944
31	21	26	30	7.500000	5.466107	5.333333	3.651641
32	22	26	31	8.000000	5.511531	5.666667	3.681692
41	30	35	40	8.500000	5.866508	6.000000	3.916763
63	51	57	62	9.000000	6.483072	6.333333	4.325820
65	53	58	64	9.500000	6.527980	6.666667	4.355643
74	61	67	73	10.000000	6.714376	7.000000	4.479467
127	113	120	126	10.500000	7.491541	7.333333	4.996243

Table 4: Codes over Z_4 and Z_8 with $D_L = 5$.

$D_L = 7$ $(d_0 = 7, d_1 = 4, d_2 = 2)$							
n	k_0	k_1	k_2	Codes over $Z_4(C_0, C_1)$		Codes over Z_8 (C_0, C_1, C_2)	
				R	L. Bound(09)	R	L. Bound(09)
7	1	3	6	5.000000	4.438258	3.666667	3.055806
8	1	4	7	5.500000	4.722507	4.000000	3.234058
11	2	6	10	7.000000	5.404182	5.000000	3.666432
15	5	10	14	7.500000	6.072010	5.333333	4.095038
16	5	11	15	8.000000	6.211164	5.666667	4.184943
23	12	17	22	8.500000	6.994609	6.000000	4.694022
27	14	21	26	9.500000	7.341113	6.666667	4.920500
30	16	24	29	10.000000	7.568876	7.000000	5.069729
32	17	26	31	10.500000	7.708415	7.333333	5.161282
35	20	28	34	11.000000	7.902192	7.666667	5.288570
47	31	40	46	11.500000	8.539815	8.000000	5.708468
63	46	56	62	12.000000	9.173668	8.333333	6.127180
64	46	57	63	12.500000	9.207741	8.666667	6.149718
67	49	59	66	13.000000	9.306855	9.000000	6.215293
72	53	64	71	13.500000	9.462582	9.333333	6.318369
86	66	78	85	14.000000	9.847039	9.666667	6.573054
128	107	120	127	14.500000	10.707574	10.000000	7.144007

Table 5: Codes over Z_4 and Z_8 with $D_L = 7$.

7. References

[01] Nogushi,T. et al., "6 GHz 135 Mbit/s Digital Radio System with 64 QAM Modulation," ICC'83, pp. F 2.4.1-F 2.4.6, June 1983.

[02] Pessoa,A.C.F., Pimentel,H.B., Borelli,W.C., "Esquema de Codificação Transparente à Ambiguidade de Fase para Sistemas Radio Digital 64 QAM," TELEMO'92/SBT'92, pp. 85-89, Julho, 1992.

[03] Nakamura,K., "A Class of Error Correcting Codes for DPSK Channels," ICC'79, pp. 45.4.1-45.4.6, June 1979.

[04] Berlekamp,E.R., "Negacyclic Codes for the Lee Metric," Combinatorial Mathematics, Chapel-Hill, NC: Univ. North Carolina, ch. 17, pp. 98-316, 1969.

[05] Chiang,J.C., Wolf,J.K., "On Channels and Codes for the Lee Metric," Inform. Control, vol. 19, pp. 159-173, 1971.

[06] Satyanarayana,C., "Lee Metric Codes over Integer Residue Rings," IEEE Trans. Info. Theory, vol. IT-25, pp. 250-254, March 1979.

[07] Baldini F.,R., "Códigos Pseudo-Cíclicos Multiníveis," SBT'89, pp. 402-405, Florianópolis, SC, Setembro, 1989.

[08] Shankar,P., "On BCH Codes over Arbitrary Integer Rings," IEEE Trans. Info. Theory, vol. IT-25, pp. 480-483, July 1979.

[09] Blake,I.F., "Codes over Certain Rings," Inform. Contr., vol. 20., pp. 396-404, 1972.

[10] Blake,I.F., "Codes over Integer Residues Rings," Inform. Contr., vol. 29, pp. 295-300, 1975.

[11] Spiegel,E., "Codes over Z_m," Inform. Contr., vol. 35, pp. 48-52, 1977.

[12] MacWilliams,F.J., Sloane,N.J.A., *The Theory of Error-Correcting Codes*

[13] Weldon Jr.,E.J., "Decoding Binary Block Codes on Q-ary Output Channels," IEEE Trans. Info. Theory, vol. IT-17, pp. 713-718, November 1971.

[14] Forney Jr.,G.D., "Generalized Minimum Distance Decoding," IEEE Trans. Info. Theory, vol. IT-12, pp. 125-131, April 1966.

[15] Yu,C.C.H., Costello Jr.,D.J., "Generalized Minimum Distance Decoding Algorithms for Q-ary Output Channels," IEEE Trans. Info. Theory, vol. IT-26, pp. 238-243, March 1980.

[16] Rudolph,L.D. et al., "Algebraic Analog Decoding of Linear binary Codes," IEEE Trans. Info. Theory, vol. IT-24, pp. 430-440, July 1979.

[17] Verhoeff,T., "An Update Table of Minimum-Distance Bounds for Binary Linear Codes," IEEE Trans. Info. Theory, vol. IT-33, pp. 665-680, September 1987.

[18] Kschischang,F.R., Pasupathy,S., "Some Ternary and Quaternary Codes and Associated Sphere Packings," IEEE Trans. Info. Theory, vol. IT-38, pp. 227-246, March 1992.

Properties of the Fourier Transform Distilled from Algebraic-Geometry Codes

Richard E. Blahut

IBM Owego NY

Abstract: The discrete Fourier transform is of such fundamental importance that it is appropriate to keep a dictionary of its elementary properties. In this talk we state several properties of the two-dimensional discrete Fourier transform that can be distilled from recent work on the decoding of algebraic-geometry codes.

I. Introduction

The discrete Fourier transform is of fundamental importance in the study of signals. Likewise, the two-dimensional discrete Fourier transform is fundamental in the study of images. It is always a good policy to be familiar with all of the ways that a tool can be used; hence it is appropriate to keep a dictionary of the elementary properties of the two-dimensional discrete Fourier transform. The purpose of this talk is to distill and formalize several properties that arise in the context of decoding algorithms for algebraic-geometry code [1-6] especially in the work of Sakata. Most properties stated in this talk are not original here. What is new is our insistence that these properties can be presented in a general setting independent of any application.

The discrete Fourier transform in two variables exists in any field. If the field F (or an extension of F) contains elements β and γ of order n' and n" respectively, then the n' by n" array s has a two-dimensional Fourier transform S given by

$$S_{j'j''} = \sum_{i'=0}^{n'-1} \sum_{i''=0}^{n''-1} \beta^{i'j'} \gamma^{i''j''} s_{i'i''} \qquad \begin{array}{l} j'=0,\ldots,n'-1 \\ j''=0,\ldots,n''-1 \end{array}$$

We shall restrict discussion to the case where n'=n"=n. We shall also limit the setting to two dimensions; generalization to more than two dimensions is probably straightforward.

Because we quote material, without proof, that is distilled from papers on decoding block codes over finite fields, we will henceforth restrict discussion to finite fields.

II. The Number of Zeros of a Polynomial Ideal

The fundamental theorem of algebra says that a (monovariate) polynomial of degree n over a field F has at most n zeros in F (or any extension of F). We will give a weak generalization of this statement to bivariate polynomials. First we restate the fundamental theorem in the language of polynomial ideals.

Let $(p(x))$ be the ideal generated by the polynomial $p(x)$. The *leading term* of any polynomial is the nonzero term with largest index. The *leading index* of any polynomial is the index of the leading term. We need to coin a term for the set of indices that are not leading indices.

Definition The *footprint* of the ideal $(p(x))$ is the set of nonnegative integers i that are not the leading index of any polynomial in $(p(x))$.

If $p(x)$ has degree n, it is clear that the footprint of $(p(x))$ is $\{0,1,\ldots,n-1\}$.

The fundamental theorem of algebra is now the statement that the number of zeros of $p(x)$ is not larger than the cardinality of the footprint of $(p(x))$. This is the form of the theorem that we will generalize to two (or more) dimensions. But first we need to order the coefficients of a bivariate polynomial so that we can identify the leading coefficient of a polynomial.

The polynomial $p(x,y)$ has coefficients $p_{i'i''}$. We require an ordering rule on the index pairs (i',i'') that respects multiplication of polynomials by monomials. That is, we require that if $(i',i'') < (j',j'')$, then $(i'+a, i''+b) < (j'+a, j''+b)$. The two most natural orders are:

Lexicographic order: $(i',i'') < (j',j'')$ if $i' < j'$ or if $i' = j'$ and $i'' < j''$.

Graded order: $(i',i") < (j',j")$ if $i'+i" < j'+j"$ or if $i'+i" = j'+j"$ and $i' < j'$.

Either order is satisfactory for our purposes.

In the ring of polynomials in two variables, let I be any ideal. The footprint of I is the set of index pairs $(i',i")$, both nonnegative, such that $(i',i")$ is not the leading index of any polynomial in I. A zero of I is a pair of field elements (a,b) such that $p(a,b) = 0$ for every $p(x,y) \varepsilon I$.

Theorem The number of zeros of I is not larger than the cardinality of the footprint of I.

In the language of the Fourier transform, this is: The number of common zero components of the spectra of polynomials in I is not larger than the footprint of I.

III. Linear Complexity Property

The weight of a sequence is the number of nonzero components in the sequence. The linear complexity of a sequence is the length of the shortest linear recursion that will generate that sequence. The one-dimensional linear complexity property [7] (sometimes called Blahut's theorem [8]) is as follows:

Theorem The weight of a finite sequence is equal to the linear complexity of its Fourier transform cyclically repeated.

We will extend this to two dimensions, first defining weight and linear complexity in a way that leads to the generalizing theorem. The weight of an n' by n" array is defined as the number of nonzero components of the array.

A locator polynomial for the array $S_{j'j''}$ represented by the polynomial

$$S(x,y) = \sum_{j'=0}^{n'-1} \sum_{j''=0}^{n''-1} S_{j'j''} x^{j'} y^{j''}$$

is a nonzero polynomial $\Lambda(x,y)$ that satisfies

$$\Lambda(x,y)S(x,y) = 0 \quad (\text{mod } x^{n'}-1)\ (\text{mod } y^{n''}-1)$$

Obviously, if $\Lambda(x,y)$ is a locator polynomial for the array S, then any polynomial multiple of $\Lambda(x,y)$ is also a locator polynomial for S.

(The coefficients of the locator polynomial can be interpreted as the tap weights of a two-dimensional linear feedback shift register.)

Definition The locator ideal for the array S is

$$\boldsymbol{\Lambda}(S) = \{\Lambda(x,y) \mid \Lambda(x,y) \text{ is a locator polynomial for } S\}$$

We may also speak of locator polynomials and locator polynomials for which the coefficients of $\Lambda(x,y)S(x,y)$ are required to be zero only for the first r coefficients in the chosen order.

Definition The linear complexity $L(S)$ of the array S is equal to the cardinality of the footprint of the locator ideal of S.

Theorem The weight of a two-dimensional array S is equal to the linear complexity of its two-dimensional Fourier transform cyclically repeated.

IV. Massey's Theorem

Massey's theorem is an inequality relationship between the length of a minimum-length linear feedback shift register that generates a sequence and the length of a minimum-length linear feedback shift register that generates a proper subsequence of that sequence.

Massey's Theorem If the minimum-length linear feedback shift register that generates $(S_0,S_1,\ldots S_{r-2})$ does not generate $S = (S_0,S_1,\ldots,S_{r-2},S_{r-1})$, then $L(S) \geq \max\,[t,r-t]$.

To say that the shift register $(\Lambda(x),t)$ generates $(S_0,S_1,\ldots S_{r-2})$ is another way of saying that the recursion

$$S_i = - \sum_{j=1}^{t} \Lambda_j S_{i-j}$$

is satisfied for $i = t+1,\ldots,r-2$.

To state Massey's theorem for the case of two variables, we will let single indices denote index pairs in the chosen total order. Thus $j=(j',j'')$ and so forth. The elements of the array S can then be put in order $S_0, S_1, S_2, S_3, \ldots$

Massey's Theorem (In two dimensions) If $\Lambda(x,y)$ is such that the jth coefficient of $\Lambda(x,y)S(x,y)$ equals zero for $j=0,\ldots,r-2$ and the $(r-1)$th coefficient of $\Lambda(x,y)S(x,y) \neq 0$, then the footprint of the locator ideal of $(S_0, S_1, \ldots, S_{r-2}, S_{r-1})$ contains the point $r-t$ where t is the leading index of $\Lambda(x,y)$.

References

1. Sakata, S., "Finding a Minimal Set of Linear Recurring Relations Capable of Generating a Given Finite Two-Dimensional Array," Journal of Symbolic Computation, Volume 5, pp 321-337, 1988

2. Sakata, S., "Decoding Binary 2-D Cyclic Codes by the 2-D Berlekamp-Massey Algorithm," IEEE Transactions on Information Theory, Volume 37, pp 1200-1203, 1991

3. Sakata, S., "A Groebner Basis and a Minimum Polynomial Set of a Finite n-D Array," in Applied Algebra, Algebraic Algorithms, and Error-Correcting Codes: Proceedings of AAECC-8, Tokyo, 1990, (S. Sakata - Editor) Springer-Verlag, Berlin, 1991

4. Massey, J. L., "Shift-Register Synthesis and BCH Decoding," IEEE Transactions in Information Theory, Vol 15, pp 122-127, 1969

5. Saints, K., and C. Heegard, "On Hyperbolic Cascaded Reed-Soloman Codes," Proceedings of the Tenth International Symposium on Applied Algebra, and Error-Correcting Codes, San Juan, Puerto Rico, 1993

6. Justesen, J., K. J. Larsen, H. E. Jensen, and T. Hoholdt "Fast Decoding of Codes from Algebraic Plane Curves," IEEE Transactions on Information Theory, Vol 38, pp 111-119, 1992

7. Blahut, R. E., Algebraic Methods for Signal Processing and Communication Coding, Springer-Verlag, 1992

8. Schaub, T., A Linear Complexity Approach to Cyclic Codes, Doctor of Technical Sciences Dissertation, Eth Swiss Federal Institute of Technology, 1988

d-Form Sequences: Families of Sequences with Low Correlation Values and Large Linear Span

Andrew Klapper*
Department of Computer Science
915 Patterson Office Tower
University of Kentucky
Lexington, KY 40506-0027
E-mail: klapper@ms.uky.edu

September 24, 1993

Abstract

Large families of binary sequences with low correlation values and large linear span are critical for spread spectrum communication systems. In this paper we described a method for constructing such families from families of homogeneous functions over finite fields satisfying certain properties. We then use this general method to construct specific families of sequences with optimal correlations and exponentially better linear span than No sequences.

1 Introduction

The volume of communication traffic over a variety of media has been steadily increasing over the past few decades, and will continue to do so. This increase has led to a need for methods that allow many users to share communication channels. Among proposed methods for such sharing, Code Division Multiple Access holds great promise, in part due to its ability to resist interference from hostile agents [16]. In order for this promise to be met, however, it is essential to find large families of easily generated binary sequences with high linear spans and low correlation function values. The smaller the pairwise cross-correlations and the larger the family, the higher the capacity of the system. Also, the higher the linear span, the harder it is for an adversary to jam or intercept messages.

Unfortunately, there are a limited number of known instances of such sequences. The correlation properties of sequences generated by various modified shift registers have been studied, including GMW sequences, [3], geometric sequences [8], cascaded GMW sequences [1, 9]. These results are often closely related to results from coding theory, as

---

*Project sponsored by the Natural Science and Engineering Research Council under Grant Number OGP0121648 and the National Security Agency under Grant Number MDA904-91-H-0012. The United States Government is authorized to reproduce and distribute reprints notwithstanding any copyright notation hereon.

Family	n	Size of Family	Maximum Correlation	Maximum Linear Span	Range of Imbalance
Gold	$2m+1$	2^n+1	$1+2^{\frac{n+1}{2}}$	$2n$	$[1, 2^{\frac{n+1}{2}}+1]$
Gold	$4m+2$	2^n-1	$1+2^{\frac{n+2}{2}}$	$2n$	$[1, 2^{\frac{n+2}{2}}+1]$
Kasami (Small Set)	$2m$	$2^{\frac{n}{2}}$	$1+2^{\frac{n}{2}}$	$\frac{3n}{2}$	$[1, 2^{\frac{n}{2}}+1]$
Kasami (Large Set)	$4m+2$	$2^{\frac{n}{2}}(2^n+1)$	$1+2^{\frac{n+2}{2}}$	$\leq \frac{5n}{2}$	$[1, 2^{\frac{n+2}{2}}+1]$
Bent	$4m$	$2^{\frac{n}{2}}$	$1+2^{\frac{n}{2}}$	$\binom{n/2}{n/4}2^{\frac{n}{4}}$	1
No	$2m$	$2^{\frac{n}{2}}$	$1+2^{\frac{n}{2}}$	$m(2^m-1)$	$[1, 2^{\frac{n}{2}}+1]$
TN	$2km$	$2^{\frac{n}{2}}$	$1+2^{\frac{n}{2}}$	$> 3mk(3k-1)^{m-2}$	$[1, 2^{\frac{n}{2}}+1]$

Table 1: Comparison of Properties of Families of Sequences of Period 2^n-1

in the case of -1 decimations of m-sequences [17]. In a few cases families of sequences with good correlation properties have been found, such as Kasami sequences [4, 5], bent function sequences [10, 11, 14, 15], and No sequences [13]. Table (1) summarizes the properties of some of these families.

In this paper we present a general method for constructing families of sequences with low cross-correlation values from families of homogeneous functions (d-forms if the degree is d) with certain properties. The families of sequences that arise include No sequences [13]. We then exhibit a particular family of quadratic forms with the required property, and show that the linear spans of the resulting sequences, trace norm (or TN) sequences, are larger (in some cases, asymptotically exponentially larger) than the linear spans of No sequences. Moreover, we describe a method for implementing TN sequences that shows that they are no more difficult to generate than No sequences. The proofs of most results are omitted in this abstract.

We generate d-form sequences in three steps. Let $q = 2^m$ be a power of two and n be a positive integer. Our construction is based on the finite fields $GF(q)$ and $GF(q^n)$. We start with a sequence of powers of a primitive element in $GF(q^n)$. To this sequence we apply a function H mapping to $GF(q)$. We then raise the result to some power. Finally, we apply a trace function mapping to $GF(2)$. The function H must be given by a "d-form" on $GF(q^n)$ over $GF(q)$ – a homogeneous function of degree d. That is, H is a d-form, if for any $x \in GF(q^n)$ and $y \in GF(q)$, we have

$$H(yx) = y^d H(x).$$

The precise definition of d-form sequences is as follows.

Definition 1.1 *Let n and m be a positive integers, and let $q = 2^m$. Let r and d be positive integers such that $\gcd(r, q-1) = \gcd(d, q-1) = 1$. Let α be a primitive element in $GF(q^n)$. Let $H(x)$ be a d-form on $GF(q^n)$ over $GF(q)$. Then the sequence* **S** *whose ith term is*

$$s_i = Tr_1^m((H(\alpha^i))^r) \tag{1}$$

*is called a d-*form sequence.

Here Tr_1^m is the trace function from $GF(2^m)$ to $GF(2)$, $Tr_1^m(x) = \sum_{i=0}^{m-1} x^{2^i}$.

2 Cross-Correlations of d-form Sequences

Recall that the cross-correlation with shift τ, of two sequences $\mathbf{S} = (s_1, s_2, \cdots)$ and $\mathbf{T} = (t_1, t_2, \cdots)$ of period N, is defined by

$$\Theta_{\mathbf{S},\mathbf{T}}(\tau) = \sum_{i=1}^{N} (-1)^{s_i + t_{i+\tau}}. \tag{2}$$

In our case $N = q^n - 1$. Our first result is a complete description of the cross-correlations of d-form sequences in terms of the zeros of d-forms.

Theorem 2.1 *Let the integers m, n, and r, and primitive element $\alpha \in GF(q^n)$ be fixed, and let H_1 and H_2 be d-forms on $GF(q^n)$ over $GF(q)$ defining d-form sequences $\mathbf{S}^1$ and $\mathbf{S}^2$, as in equation (1). For any shift τ, let $z_\tau = |\{x \neq 0 \in GF(q^{2k}) : H_1(x) + H_2(\alpha^\tau x) = 0\}|$. Then*

$$\Theta_{\mathbf{S}^1,\mathbf{S}^2}(\tau) = \frac{qz_\tau - (q^n - 1)}{q - 1}.$$

If $d = 2$, and H is a quadratic form, then the number of zeros of H is well understood. Recall that the *rank* of a quadratic form H is the smallest integer t such that there is a set of coordinates in which H can be represented using only t variables. The number of solutions x to the equation $H(x) = 0$ (or, more generally, $H(x) = a$) is determined by the rank and, in the case of even rank, whether H is of one of two types. A nice treatment of this analysis can be found in Lidl and Neiderreiter's book [12].

Corollary 2.2 *Let H_1 and H_2 be quadratic forms defining d-form sequences $\mathbf{S}^1$ and $\mathbf{S}^2$, and let τ be any integer. If the rank of $H_1(x) + H_2(\alpha^\tau x)$ is t, then*

$$\Theta_{\mathbf{S}^1,\mathbf{S}^2}(\tau) = -1$$

when t is odd, and

$$\Theta_{\mathbf{S}^1,\mathbf{S}^2}(\tau) = \pm q^{n-t/2} - 1$$

when t is even. In particular, if for every τ the rank is n or odd, then the cross-correlations of $\mathbf{S}^1$ and $\mathbf{S}^2$ are three valued with values in $\{-q^{n/2} - 1, -1, q^{n/2} - 1\}$.

Proof: The corollary follows from the fact that a quadratic form with odd rank takes value zero q^{n-1} times, while a quadratic form with even rank t takes the value zero $q^{n-1} \pm (q-1)q^{n-t/2-1}$ times [12]. □

Corollary 2.3 *If $\mathcal{F}$ is a family of quadratic forms on $GF(q^n)$ over $GF(q)$ such that for any H_1 and H_2 in $\mathcal{F}$, and integer τ, the rank of the quadratic form $H_1(x) + H_2(\alpha^\tau x)$ is n or odd, then $\mathcal{F}$ defines a family of quadratic sequences with three valued cross-correlations with values in $\{-q^{n/2} - 1, -1, q^{n/2} - 1\}$.*

3 Trace Norm (TN) Sequences

In this section we describe a class of families of quadratic form sequences that achieve the cross-correlations of Corollary 2.3 and have large linear span.

Definition 3.1 *Let m and k be positive integers, and let $q = 2^m$. Let r be a positive integer such that* $\gcd(r, q-1) = 1$. *Let α be a primitive element in $GF(q^{2k})$ and let γ be an element of $GF(q^k)$. Let $T = q^k + 1$. Then the sequence* $\mathbf{S}$ *whose ith term is*

$$s_i = Tr_1^m((Tr_m^{mk}(Tr_{mk}^{2mk}(\alpha^{2i}) + \gamma\alpha^{iT}))^r) \tag{3}$$

is a Trace Norm (or TN) sequence.

The function x^T is the norm function from $GF(q^n)$ to $GF(q)$. We will be interested in families of TN sequences with all parameters besides γ fixed. In this case, we will denote the TN sequence with parameter γ by $\mathbf{S}^\gamma$.

Theorem 3.2 *Let $\mathbf{S}^\gamma$ and $\mathbf{S}^\delta$ be two TN sequences, based on the same integers m, k, and r. Then the cross-correlations of $\mathbf{S}^\gamma$ and $\mathbf{S}^\delta$ are three valued, with values in $\{-q^k - 1, -1, q^k - 1\}$ unless $\mathbf{S}^\gamma = \mathbf{S}^\delta$ and $\tau = 0$.*

Proof sketch: By Theorem 2.1, it suffices to determine the number of zeros of the function

$$Tr_m^{mk}(Tr_{mk}^{2mk}((1 + \alpha^{2\tau})x^2) + (\gamma + \delta\alpha^{T\tau})x^T),$$

which is a quadratic form on $GF(q^{2k})$ over $GF(q)$. The number of zeros of such a quadratic form can be counted using the results of [7]. □

The imbalance $I(\mathbf{S})$ of a binary sequence $\mathbf{S}$ is the number of zeros in $\mathbf{S}$ minus the number of ones. The proof of the Theorem 3.2 can be used to find the imbalance of a TN sequence.

Proposition 3.3 *The imbalance of a TN sequence is -1, $-q^k - 1$, or $q^k - 1$.*

There are six possible distributions of values of the cross-correlation of two TN sequences.

Proposition 3.4 *Let $\mathbf{S}^\gamma$ and $\mathbf{S}^\delta$ be two TN sequences based on the same field parameters m and k and exponent r. The distribution of values of the cross-correlation of $\mathbf{S}^\gamma$ and $\mathbf{S}^\delta$ is given by one of the following cases:*

1. *$q^{2k} - 1$ occurs once and -1 occurs $q^{2k} - 2$ times.*
2. *$q^k - 1$ occurs $(q^{2k} + q^k)/2 - 1$ times and $-q^k - 1$ occurs $(q^{2k} - q^k)/2$ times.*
3. *$q^k - 1$ occurs $(q^{2k} + q^k)/2$ times and $-q^k - 1$ occurs $(q^{2k} - q^k)/2 - 1$ times.*
4. *-1 occurs $q^k + 1$ times, $q^k - 1$ occurs $(q^{2k} + q^k)/2$ times, and $-q^k - 1$ occurs $(q^{2k} - 3q^k)/2 - 3$ times.*
5. *-1 occurs $q^k + 1$ times, $q^k - 1$ occurs $(q^{2k} + q^k)/2 - 2$ times, and $-q^k - 1$ occurs $(q^{2k} - 3q^k)/2$ times.*
6. *-1 occurs $q^k + 1$ times, $q^k - 1$ occurs $(q^{2k} - q^k)/2 - 1$ times, and $-q^k - 1$ occurs $(q^{2k} - q^k)/2 - 1$ times.*

4 Linear Span of TN Sequences

In this section we show that the linear span of a TN sequence is at least that of a GMW sequence with the same period, and that for q large enough and $k = 2$, it exceeds the linear span of a No sequence with the same period. The development is similar to that in the case of No sequences, with additional complication due to the additional trace function.

Key [6] showed that if we express the ith term of a sequence $\mathbf{S}$ as a polynomial in α^i, then the linear span $\lambda_{\mathbf{S}}$ is the number of monomials in the polynomial. That is, we must count the monomials in the polynomial

$$\begin{aligned} s(x) &= Tr_1^m((Tr_m^{mk}(Tr_{mk}^{2mk}(x^2) + \gamma x^T))^r) & (4)\\ &= Tr_1^m((Tr_m^{mk}(x^2 + x^{2q^k} + \gamma x^T))^r) & (5)\\ &= Tr_1^m((Tr_m^{mk}(x^2(1 + \gamma y + y^2)))^r), & (6) \end{aligned}$$

where $y = x^{q^k - 1}$. Expanding the trace functions, we see

$$s(x) = \sum_{j=0}^{m-1}(\sum_{i=0}^{k-1} x^{2^{j+1}q^i}(1 + \gamma y + y^2)^{2^j q^i})^r. \qquad (7)$$

It can be seen that distinct terms from the outer sum have distinct degree monomials. This implies that

$$\lambda_{\mathbf{S}} = m \cdot \left| \left\{ \text{monomials in } \left(\sum_{i=0}^{k-1} x^{2q^i}(1 + \gamma y + y^2)^{q^i} \right)^r \right\} \right|. \qquad (8)$$

The rth power in equation (8) can be expanded as

$$\sum_{\bar{a}} \prod_{i=1}^{k} (x^{2q^i}(1 + \gamma y + y^2)^{q^i})^{a_i} = \sum_{\bar{a}} x^{2\sum a_i q^i}(1 + \gamma y + y^2)^{\sum a_i q^i}, \qquad (9)$$

where the sum is over all $\bar{a} = (a_0, \cdots, a_{k-1})$ such that $0 \leq a_i$, the bitwise "AND" of any two a_is is zero, and the bitwise "OR" of all the a_i equals r. That is, the bits of r are distributed among the a_i, maintaining their relative bit positions. In particular, $a_i \leq r < q$. Reducing exponents modulo $q^k - 1$, we get monomials of degree $2\sum_i a_i q^i$. Each such expression is twice the unique base q representation of an integer, hence these monomials are pairwise distinct. Thus we have the following proposition.

Proposition 4.1 *The linear span of the TN sequence* $\mathbf{S}$ *is given by*

$$\lambda_{\mathbf{S}} = m \cdot \sum_{\bar{a}} |\{ \text{monomials in } (1 + \gamma y + y^2)^{\sum a_i q^i} \}|,$$

where the sum is over all $\bar{a} = (a_0, \cdots, a_{k-1})$ *such that* $0 \leq a_i$*, the bitwise "AND" of any two* a_i*s is zero, and the bitwise "OR" of all the* a_i *equals* r.

The cardinalities in the sums in this proposition have been evaluated by No and Kumar [13] in describing the linear span of No sequences as follows. Suppose $t < q^k/2$ is a positive integer with R runs of ones, of lengths $L_1, \cdots, L_R$ and $\gamma \neq 0$ in its base two expansion[1]. Let $\epsilon = -1$ if the quadratic $y^2 + \gamma y + 1$ is reducible over $GF(q^k)$ (that is, if $Tr_1^{mk}(1/\gamma) = 0$) and $\epsilon = 1$ otherwise. For a fixed primitive element $\alpha \in GF(q^{2k})$, there is a unique (principal) root $\delta = \alpha^b$ of the quadratic $y^2+\gamma y+1$ such that $0 \leq b < (q^{2k}-1)/2$. When $\epsilon = -1$, we have $b = c(q^k + 1)$, and when $\epsilon = 1$, we have $b = c(q^k - 1)$. Thus $0 \leq c \leq q^k/2$. Let $g = \gcd(c, q^k + \epsilon)$. Then the number of terms in $(y^2 + \gamma y + 1)^t$ is

$$\prod_{j=1}^{R} \left\{ 2^{L_j+1} - 1 - 2\left\lfloor \frac{(2^{L_j} - 1)g}{q^k + \epsilon} \right\rfloor \right\}. \tag{10}$$

When $\gamma = 0$, the number of terms is $2^{\mathrm{wt}(t)}$, where $\mathrm{wt}(t) = \sum_j L_j$ is the number of ones in the binary expansion of t.

Combining this with Proposition 4.1, we have

Theorem 4.2 *Let* $\mathbf{S}$ *be a TN sequence based on fields* $GF(q)$ *and* $GF(q^k)$, *exponent* r, *and coefficient* $\gamma \in GF(q^k)$. *When* $\gamma \neq 0$, *let* $\epsilon = -1$ *if the quadratic* $y^2 + \gamma y + 1$ *is reducible over* $GF(q^k)$ *and* $\epsilon = 1$ *otherwise. Let* c *be such that*

$$\delta = \alpha^{c(q^k+1)} \qquad \textit{when } \epsilon = -1$$

and

$$\delta = \alpha^{c(q^k-1)} \qquad \textit{when } \epsilon = 1$$

is a root of $y^2 + \gamma y + 1$, *with* $0 \leq c \leq q^k/2$. *Let* $g = \gcd(c, q^k + \epsilon)$.

For each $\bar{a} = (a_0, \cdots, a_{k-1})$ *such that* $0 \leq a_i$, *the bitwise "AND" of any two* a_i*s is zero, and the bitwise "OR" of all the* a_i *equals* r, *let* $R_{\bar{a}}$ *be the number of runs of ones in* $\sum a_i q^i$, *and let* $L_{\bar{a},1}, \cdots$ *be the lengths of the runs. Then the linear span of* $\mathbf{S}$ *is given by*

$$\lambda_{\mathbf{S}} = m \cdot \sum_{\bar{a}} \prod_{j=1}^{R_{\bar{a}}} \left\{ 2^{L_{\bar{a},j}+1} - 1 - 2\left\lfloor \frac{(2^{L_{\bar{a},j}} - 1)g}{q^k + \epsilon} \right\rfloor \right\}.$$

If $\gamma = 0$, *then the linear span of* $\mathbf{S}$ *is given by*

$$\lambda_{\mathbf{S}} = m \cdot \sum_{\bar{a}} 2^{\mathrm{wt}(r)}.$$

We next want to determine how the linear span can be maximized. First observe that if we choose γ so that

$$\gcd(c, q^k + \epsilon) < \frac{q^k + \epsilon}{q - 1}$$

then all terms involving the floor will disappear, and the linear span will be

$$\lambda_{\mathbf{S}} = m \cdot \sum_{\bar{a}} \prod_{j=1}^{R_{\bar{a}}} \left\{ 2^{L_{\bar{a},j}+1} - 1 \right\}.$$

---

[1] It should be noted that the necessary assumption that $t < q^k/2$ was not explicitly stated by No and Kumar. It poses no difficulty for their analysis, however, since any exponent over $GF(q^k)$ which is relatively prime to $q^k - 1$ can be assumed to be less than $q^k - 1$. Such an exponent thus has at least one zero in the first mk bits of its base two expansion. Since No sequences are invariant under cyclic shift of the exponent r, we can assume the exponent has a zero as its high bit. In the more general case of TN sequences, we can also cyclically shift r so its high bit is zero, and therefore the high bit of each $\sum_i a_i q^i$ is zero.

This can be done independently of the choice of r (say by choosing c relatively prime to $q^k + \epsilon$), so the maximum value of the linear span occurs with such a choice.

Next observe that the effect on an $\bar{a}$ of increasing the number of ones in the binary expansion of r is either to increase the length of a run of ones in $\sum_i a_i q^i$ by one, or to merge a run of length L and a run of length K into a run of length $L + K + 1$. In either case, the contribution to the linear span is increased, since

$$(2^{L+1} - 1) < (2^{L+2} - 1) \qquad \text{and} \qquad (2^{L+1} - 1)(2^{K+1} - 1) < (2^{K+L+2} - 1).$$

Thus the linear span is maximized by maximizing $\mathrm{wt}(r)$. We have $r < q - 1$, so $\mathrm{wt}(r)$ is maximized at $m - 1$, that is, when r has one zero and $m - 1$ ones in its binary expansion. Since the sequence, and hence the linear span, is independent of cyclic shifts of r, the maximal linear complexity occurs when r is a string of ones followed by a single zero. That is $r = 2^{m-1} - 1$.

We next estimate the linear span for these r by recursively estimating the linear span for the exponent r_i which consists of i ones followed by $m - i$ zeros. If $i > 0$, we can produce r_i from r_{i-1} by replacing the first zero with a one. Each $\bar{a}'$ for r_{i-1} gives rise to k $\bar{a}$'s for r_i, depending on which a_i receives the new bit. For $k - 1$ of them, we are adding a run of length 1, hence multiplying the contribution to the linear span by 3. For the remaining $\bar{a}$, we are increasing the length of a run by 1, and this run has length at most $i - 1$. Thus the contribution of this term is

$$\eta = \frac{2^{t+1} - 1}{2^t - 1}$$

times the contribution of the original term, for some t such that $2 \leq t < i$. Therefore $2 < \eta \leq 7/3$. Also, when r has a single one in its binary expansion, the linear span is exactly $3k$. It follows that the linear span satisfies

$$3mk(3k - 1)^{m-2} < \lambda_{\mathrm{S}} < m(3k)^2(3k - \frac{2}{3})^{m-3}.$$

For large enough $n = 2mk$, the linear complexity is maximized by taking $k = 2$, and can be made at least

$$\frac{3n}{2} 5^{n/4-2}.$$

We have shown experimentally that the base of 5 in this expression in fact gets close to 5.24 as n increases, and we can come closer to this value theoretically by improving the estimates for r_i for small i. By comparing the values achieved, we find that for $n \geq 44$ the linear span is maximized by taking $k = 2$. For smaller n, experimental data show that the linear span is maximized by taking $k = 1$, i.e., by No sequences. The linear span in this case is at most

$$\frac{n}{2}(2^{n/2+1} - 3 \cdot 2^{n/4} + 1).$$

Theorem 4.3 *The maximum possible linear span for a TN sequence is achieved by taking* $k = 2$ *and* $r = 2^{n/4-1} - 1$ *when* $n \geq 44$, *and by taking* $k = 1$ *and* $r = 2^{n/2-1} - 1$ *when* $n < 44$.

In general, we see that the linear span of TN sequences can be the same as or greater than that of No sequences while performing the exponentiation in a smaller field, with a smaller exponent r. This leads to more efficient implementation of generators of the sequences, as discussed in the next section.

5 Implementation of TN Sequence Generators

Consider the TN sequence

$$\begin{aligned} s_i^\gamma &= Tr_1^m((Tr_m^{mk}(Tr_{mk}^{2mk}(\alpha^{2i}) + \gamma\alpha^{iT}))^r) & (11) \\ &= Tr_1^m((Tr_m^{2mk}(\alpha^{2i}) + Tr_m^{mk}(\gamma\alpha^{iT}))^r) & (12) \end{aligned}$$

and let $n = 2mk$. Since α is a primitive element in $GF(q^{2k})$, α^2 is also a primitive element in $GF(q^{2k})$ and $\beta \stackrel{def}{=} \alpha^T$ is a primitive element in $GF(q^k)$. It follows that the sequence

$$Tr_m^{2mk}(\alpha^{2i})$$

is an m-sequence of elements in $GF(q)$, and can be generated by a linear feedback shift register (LFSR) of length n/m over $GF(q)$. That is, the elements of the register are elements of $GF(q)$, and the feedback function is a linear function in n/m variables. Such a register requires only n bits of storage. The arithmetic required is at most n/m multiplications by constants in $GF(q)$ (the coefficients of the minimum polynomial of α^2, some of which may be zero), and at most $n/m - 1$ additions in $GF(q)$. The arithmetic can be minimized if α is chosen to minimize the number of nonzero coefficients in its minimal polynomial.

Similarly, the sequence

$$Tr_m^{mk}(\gamma\alpha^{iT}) = Tr_m^{mk}(\gamma\beta^i)$$

is an m-sequence over $GF(q)$ which can be generated by a LFSR of length $n/(2m)$ over $GF(q)$. This requires $n/2$ bits, and at most $n/(2m)$ multiplications and $n/(2m) - 1$ additions in $GF(q)$, which can be minimized by choosing α to minimize the number of nonzero coefficients in the minimal polynomial of β. Thus the total amount of $GF(q)$ arithmetic required to implement a TN sequence is minimized by choosing α so the total number of nonzero coefficients in the minimal polynomials of α and β is minimized.

One extra addition is required to combine the outputs of the two LFSRs. The result is then raised to the rth power, and the trace to $GF(2)$ computed. However, in representing elements of $GF(q)$ as m-bit vectors, we can choose a basis so that the trace of an element is always given by projection onto a fixed component, say the first. Thus we only need to compute a single bit of the rth power.

The different choices of γ correspond to different initial loadings of the second LFSR. Thus an entire family of TN sequences can be implemented by a single hardware circuit. Changing to a new sequence is possible by simply resetting the initial loading of the second LFSR.

6 The Number of Distinct Families of TN Sequences

It is useful to know how many distinct families of TN sequences there are with the parameters $n = 2mk$, and thus the period, fixed. By showing that each choice of parameters m, k, r (up to multiplication by a power of 2), and α (up to raising to an exponent which is a power of two) gives rise to a distinct family of TN sequences, in the sense that no sequence in one family is a cyclic shift of a sequence in another family, we prove

Theorem 6.1 *Let $n = 2\ell$. The number $\mathbf{N}_{\mathrm{TN}}$ of distinct families of TN sequences of period $2^n - 1$ is given by*

$$\mathbf{N}_{\mathrm{TN}} = \frac{\phi(2^n - 1)}{n} \cdot \left(\sum_{m|\ell} \left(\frac{\phi(2^m - 1)}{m} - 1 \right) + 1 \right),$$

where $\phi(\cdot)$ is Euler's phi function.

7 Complete Families of Degree d d-Form Sequences

In this section we consider families of d-form sequences in which the fields $GF(q)$ and $GF(q^n)$, exponent r, the degree d, and the primitive element α are fixed, but the d-form $H(x)$ is free to vary through all nonzero polynomials of degree d on $GF(q^n)$ over $GF(q)$. Such a family will be called a *complete families of degree d d-form sequences.* We restrict our attention to the case when $n = 2$. Note that for $d = 2$, this gives a family of No sequences. In general the size N of such a family (identifying sequences that are cyclic shifts of each other) is

$$\begin{aligned} N &= \begin{cases} \frac{q^{d+1}-1}{q^2-1} & \text{if } d \text{ is odd,} \\ \frac{q^{d+1}-q}{q^2-1} + 1 & \text{if } d \text{ is even,} \end{cases} \\ &\sim q^{d-1}. \end{aligned}$$

Furthermore, we can bound the cross-correlations in a complete family of degree d d-form sequences as follows.

Theorem 7.1 *If $\mathcal{F}$ is a complete family of degree d d-form sequences, then every cross-correlation of any two sequences from $\mathcal{F}$ is contained in the set $\{-q-1, -1, q-1, 2q-1, \cdots, (d-1)q-1\}$. In particular, the cross-correlations are at most $d+1$ valued.*

References

[1] M. Antweiller and L. Bömer, "Complex sequences over $GF(p^m)$ with a two-level autocorrelation function and a large linear span," *IEEE Trans. on Info. Theory,* vol. 38, pp. 120-130, Jan. 1992.

[2] S. Golomb, *Shift Register Sequences,* Aegean Park Press: Laguna Hills, CA, 1982.

[3] B. Gordon, W. H. Mills, and L. R. Welch, "Some new difference sets," *Canad. J. Math.* vol. 14 pp. 614-625, 1962.

[4] T. Kasami, "Weight distribution formula for some classes of cyclic codes," Coordinated Science Laboratory, University of Illinois, Urbana, Tech. Rep. R-285 (AD632574), 1966.

[5] T. Kasami, "Weight distribution of Bose-Chaudhuri-Hocquenghem codes," in *Combinatorial Mathematics and its Applications.* Chapel Hill, NC: University of North Carolina Press, 1969.

[6] E. L. Key, "An Analysis of the structure and complexity of nonlinear binary sequence generators," *IEEE Trans. Info. Theory,* vol. IT-22 no. 6, pp. 732-736, Nov. 1976.

[7] A. Klapper, "Cross-correlations of geometric sequences in characteristic two," *Designs, Codes, and Cryptography*, vol. 3, pp. 347-377, 1993.

[8] A. Klapper, A.H. Chan, and M. Goresky, "Cross-correlations of linearly and quadratically related geometric sequences and GMW Sequences," in press, *Discrete Applied Mathematics.*

[9] A. Klapper, A.H. Chan, and M. Goresky, "Cascaded GMW Sequences," *IEEE Trans. on Info. Theory,* vol. IT-39, pp. 177-183, 1993.

[10] P. V. Kumar and R. A. Scholtz, "Bounds on the linear span of bent sequences," *IEEE Trans. Info. Theory,* vol. IT-29, pp. 854-862, Nov. 1983.

[11] A. Lempl and M. Cohn, "Maximal families of bent sequences," *IEEE Trans. Info. Theory,* vol. IT-28, pp. 865-868, Nov. 1982.

[12] R. Lidl and H. Niederreiter *Finite Fields* in *Encyclopedia of Mathematics vol. 20,* Cambridge University Press: Cambridge, 1983.

[13] J. No and P. V. Kumar, "A new family of binary pseudorandom sequences having optimal periodic correlation properties and large linear span," *IEEE Trans. on Info. Theory*, vol. 35, pp. 371-379, 1989.

[14] J. D. Olsen, R. A. Scholtz, and L. R. Welch, "Bent-function sequences," *IEEE Trans. Inform. Theory,* vol. IT-28, pp. 858-864, Nov. 1982.

[15] O. Rothaus, "On bent functions," *Journal of Combinatorial Theory Series A*, vol. 20, pp. 300-305, 1976.

[16] M. Simon, J. Omura, R. Scholtz, and B. Levitt, *Spread-Spectrum Communications,* Volume 1, Computer Science Press: 1985.

[17] J. Wolfmann, "New bounds on cyclic codes from algebraic curves," in *Proc. 1988 Conference on Coding Theory and Its Applications,* G. Cohen, J. Wolfmann, Eds., *Lecture Notes in Computer Science Vol. 388*, Springer-Verlag: Berlin, pp. 47-62, 1989.

Partial Period Cross-Correlations of Geometric Sequences

Andrew Klapper*
Department of Computer Science
915 Patterson Office Tower
University of Kentucky
Lexington, KY 40506-0027
E-mail: klapper@ms.uky.edu

August 24, 1993

Abstract

For a binary pseudorandom sequences $\{\mathbf{S}_i\}$ and $\{\mathbf{T}_i\}$ with period N, the partial period cross-correlation function $\Theta_{\mathbf{S},\mathbf{T}}(\tau, k, D)$ is defined by correlating the portion of $\{\mathbf{S}_i\}$ within a window of size D, and start position k, with the portion of $\{\mathbf{T}_i\}$ in another window of the same size but starting τ steps later in the sequence. A distribution of possible partial period cross-correlation values is obtained by allowing the start position k to vary over all possible values $0 \leq k < N$. The expectation value is proportional to the periodic cross-correlation function $\Theta_{\mathbf{S},\mathbf{T}}(\tau)$. In this paper the variance of the partial period cross-correlation values is estimated for a large class of binary pseudorandom sequences, the so-called "geometric sequences," with underlying m-sequences related by linear decimations.

1 Introduction

A critical step in the development of practical CDMA communication systems is the discovery of large families of sequences with small pairwise cross-correlation values. The larger the family and the smaller the cross-correlation, the greater the capacity of the system. A great deal of effort in recent years has gone into the search for such families of sequences, with some degree of success. However, the implementation of CDMA systems has been held up because small pairwise cross-correlation is not the only requirement for sequences used in these systems. In addition, it is essential

---

*Project sponsored by the Natural Science and Engineering Research Council under Grant Number OGP0121648 and the National Security Agency under Grant Number MDA904-91-H-0012. The United States Government is authorized to reproduce and distribute reprints notwithstanding any copyright notation hereon.

that their partial period cross-correlation values (partial sums of the type of sums that occur in cross-correlations) be provably low. Yet there are few results of any kind giving good bounds on these statistics (See [8] and [5] for surveys of known results up to 1985). In fact, in many cases sequences known by theoretical results to have good cross-correlations have been shown empirically to have rather bad partial period cross-correlations. There is a real need for good theoretical bounds on partial period cross-correlations. In this paper we give explicit partial period cross-correlation estimates for certain pairs of binary pseudorandom sequences, geometric sequences whose underlying m-sequences are linearly decimated, but whose feedforward functions may differ. Due to the wide variation in the values of partial period cross-correlation, the best we can hope to do is give sharp bounds on their expectation and variance, and this is what we do. More specifically, we show that for certain pairs of geometric sequences the expectation of the partial period cross-correlations is usually low, and the variance is always low. These results generalize those of Klapper and Goresky on the partial period autocorrelations of geometric sequences [7].

Geometric sequences are obtained by applying a nonlinear "feedforward function" $f : GF(q) \rightarrow GF(2)$ to a linearly recurrent sequence with values in a finite field $GF(q)$. This large class of pseudorandom sequences includes m-sequences [3], GMW sequences [4, 13], Bent sequences [11, 14], cascaded GMW sequences [6], the Chan-Games stream cipher [1] and many others. Since they are readily generated using shift register hardware, may have enormous linear span [1, 6, 12], and optimally low periodic autocorrelation values and low periodic cross-correlation values [6], geometric sequences are candidates for use in secure spread spectrum applications.

It is well known [8, 14] that the expected value of the partial period cross-correlation values for a periodic sequence is proportional to the periodic cross-correlation values. The full period cross-correlations of geometric sequences have recently been computed in the case in question [2], and in some cases are low for most shifts. Thus the same will be true for the average partial period cross-correlations. This fact is only useful, however, if we further know that the variance of the partial period cross-correlations is low. The bound we give for the variance of partial period cross-correlations does not involve any knowledge of the feedforward functions. We compute the expectation and variance of these partial period correlations based on a distribution in which the window size is kept fixed and the start position of the window is allowed to vary uniformly. This is analogous to the distribution used in [14], where the case of m-sequences was studied. Several authors who have studied similar questions average these correlation values over all possible start positions *and* all possible shifts τ. The double averaging results in a somewhat easier expression to evaluate but the resulting information may be less valuable than that which is derived here.

2 Geometric Sequences and Correlations

In this section we recall the definition of geometric sequences and some of their basic properties, and the definition of full and partial period cross-correlation functions of periodic sequences. Geometric sequences are based on algebra over finite fields, and we recall first some of the basic concepts we will use. See Lidl and Niederreiter's or McEliece's book [9, 10] for a more detailed treatment of finite fields.

Let q be a fixed power of a prime number, and let $GF(q)$ denote the Galois field with q elements. We consider this to be a "base" field. For any $n \geq 1$, we denote the *trace function* from $GF(q^n)$ to $GF(q)$ by $Tr_q^{q^n}$, defined by $Tr_q^{q^n}(x) = \sum_{i=0}^{n-1} x^{q^i}$. Then $Tr_q^{q^n}$ is a $GF(q)$-linear function. For any $m \geq 1$ we have, $Tr_q^{q^{nm}}(x) = Tr_q^{q^n}(Tr_{q^n}^{q^{nm}}(x))$.

Let α be a primitive element of $GF(q^n)$. The infinite periodic sequence $\mathbf{U}$ whose ith term is $\mathbf{U}_i = Tr_q^{q^n}(\alpha^i) \in GF(q)$ is known as an *m-sequence* over $GF(q)$ of span n [1] [9]. Throughout this paper we fix a prime power q, an integer n, and a primitive element $\alpha \in GF(q^n)$.

Definition 2.1 (Chan and Games [1]) *If $f : GF(q) \rightarrow GF(2)$ is any function, then the binary sequence $\mathbf{S}$ whose ith term is*

$$\mathbf{S}_i = f(Tr_q^{q^n}(\alpha^i))$$

is the geometric sequence based on the primitive element α and feedforward function f.

Such a geometric sequence is a binary periodic sequence whose period divides q^n-1. Geometric sequences with q even have been suggested for use in spread spectrum communication systems, due to their (in some cases) optimal autocorrelations, excellent cross-correlation values, and relatively high linear complexities. Geometric sequences with q odd have been used in applications where easily generated sequences with large linear complexities are needed. The geometric sequence $\mathbf{S}$ is easy to generate if the feedforward function f is easy to compute. If $g : GF(q) \rightarrow GF(2)$ is a second function, and $\beta = \alpha^k$ another primitive element in $GF(q^n)$, we denote by $\mathbf{T}$ the geometric sequence whose ith term is

$$\mathbf{T}_i = g(Tr_q^{q^n}(\beta^i)).$$

Throughout the rest of this paper we will assume that $k = p^e$ for some integer e. This is the so-called "linear decimation" case. Note that in this case

$$\mathbf{T}_i = g(Tr_q^{q^n}(\alpha^{ip^e})) = g((Tr_q^{q^n}(\alpha^i))^{p^e}).$$

Thus if we replace $g(x)$ by $g'(x) \overset{\text{def}}{=} g(x^{p^e})$, we may assume that the underlying m-sequences of the two geometric sequences are the same.

Definition 2.2 *The periodic cross-correlation function $\Theta_{\mathbf{S},\mathbf{T}}(\tau)$ of $\mathbf{S}$ and $\mathbf{T}$ is the function whose value at τ is the correlation of $\mathbf{S}$ with the τ-shift of $\mathbf{T}$.*

$$\Theta_{\mathbf{S},\mathbf{T}}(\tau) = \sum_{i=1}^{q^n-1} (-1)^{\mathbf{S}_i}(-1)^{\mathbf{T}_{i+\tau}}$$

We next recall a result due to Chan, Goresky, and Klapper [2] regarding the cross-correlation of a geometric sequence. We use the following notation: $F(x) = (-1)^{f(x)}$, $I(f) = \sum_{x \in GF(q)} F(x)$, the *imbalance* of f (and similarly for g and g'), and

$$\Delta_a^e(f,g) = \sum_{x \in GF(q)} F(x)G(ax^{p^e}),$$

---

[1] More generally, we may also consider the sequence whose ith term is $Tr_q^{q^n}(A\alpha^i)$ for some fixed element A of $GF(q^n)$. This amounts to a cyclic shift of the first sequence, so we do not consider it to be a distinct sequence here.

the *short cross-correlation function*[2] of f and g. Set $\nu = (q^n - 1)/(q-1)$. Then $\alpha^\tau \in GF(q^n)$ lies in the subfield $GF(q) \iff \tau$ is a multiple of ν.

Theorem 2.3 *The values for the periodic cross-correlation (with shift $\tau \neq 0$) of the geometric sequence* **S** *are:*

1. $\Theta_{\mathbf{S},\mathbf{T}}(\tau) = q^{n-2} I(f) I(g) - F(0)G(0)$, *if τ is not a multiple of ν.*
2. $\Theta_{\mathbf{S},\mathbf{T}}(\tau) = q^{n-1} \Delta^e_{\alpha^\tau}(f,g) - F(0)G(0)$, *if ν divides τ.*

Corollary 2.4 *If* **S** *is as balanced as possible, i.e. $I(f), I(g) = \pm 1$ if q is odd, and $I(f) = 0$ if q is even. Then for a shift τ that is not a multiple of ν,*

$$\Theta_{\mathbf{S},\mathbf{T}}(\tau) = \pm q^{n-2} - 1$$

if q is odd, and

$$\Theta_{\mathbf{S},\mathbf{T}}(\tau) = -1$$

if q is even.

Thus, if q is odd, the cross-correlations are high. When q is even, the feedforward functions f and g can be chosen to be balanced, and the shifted cross-correlations are low for all but at most $q-1$ shifts.

The partial period cross-correlation of a sequence is defined by limiting the range of values in the sum defining the periodic cross-correlation to a fixed window. It is parametrized by the start position k and length D of the window, as well as the shift τ.

Definition 2.5 *The partial period cross-correlation function of a periodic sequence* **S** *is*

$$\Theta_{\mathbf{S},\mathbf{T}}(\tau, k, D) = \sum_{i=k}^{D+k-1} (-1)^{\mathbf{S}_i} (-1)^{\mathbf{T}_{i+\tau}}$$

3 Expected Partial Period Cross-Correlations

We denote the *expectation* of a random variable X by $E[X]$. All expectations are taken for fixed window size D and shift τ, assuming a uniform distribution on all start positions k.

Theorem 3.1 *Suppose either* **S** *or* **T** *is as balanced as possible, i.e. the imbalance is ± 1 if q is odd, and 0 if q is even. Then for a shift τ that is not a multiple of ν, the expected partial period cross-correlation of* **S** *and* **T** *is*

$$E[\Theta_{\mathbf{S},\mathbf{T}}(\tau, k, D)] = D(q^{n-2} - 1)/(q^n - 1)$$

if q is odd, and

$$E[\Theta_{\mathbf{S},\mathbf{T}}(\tau, k, D)] = -D/(q^n - 1)$$

if q is even.

This theorem follows from the fact that, for any sequences **S** and **T** of period N,

$$E[\Theta_{\mathbf{S},\mathbf{T}}(\tau, k, D)] = \frac{D}{N} \Theta_{\mathbf{S},\mathbf{T}}(\tau).$$

---

[2] If γ is a primitive element of $GF(q)$, and $a = \gamma^\sigma$, then $\Delta^e_a(f) - F(0)G(0)$ is the cross-correlation with shift σ of the sequence whose ith term is $f(\gamma^i)$ and the sequence whose ith term is $g(\gamma^{ip^e})$.

4 Variance of Partial Period Cross-Correlations

We next consider the variance of the partial period cross-correlation. Recall that the *variance* of a random variable X is defined to be $V(X) = E[(X - E[X])^2] = E[X^2] - E[X]^2$. The variance is a measure of the extent to which a random variable remains close to its expectation.

Theorem 4.1 *For any τ, the variance of the partial period cross-correlation of a geometric sequence with shift τ and window size D is bounded above by*

$$\frac{q^n D}{q^n - 1}\left\lceil\frac{(q-1)D}{q^n - 1}\right\rceil (q^2 + q + 1).$$

If f or g is balanced, this can be reduced to

$$\frac{q^n D}{q^n - 1}\left\lceil\frac{(q-1)D}{q^n - 1}\right\rceil \frac{(q^2 + q + 2)}{2}.$$

To prove this, we must determine the second moment $E[\Theta_{\mathbf{S},\mathbf{T}}(\tau, k, D)^2]$ of the partial period cross-correlation. The proof proceeds by reducing the second moment to an expression involving sums of intersections of hyperplanes in n space over $GF(q)$, then analyzing the number of times various types of intersections occur. The field $GF(q^n)$ is an $n-$dimensional vector space over the base field $GF(q)$. For any $s \in GF(q)$ and $A \neq 0 \in GF(q^n)$, the set

$$H_A^s = \{x \in GF(q^n) : Tr_q^{q^n}(Ax) = s\}$$

is an affine hyperplane in $GF(q^n)$, i.e. a translate of an $(n-1)$-dimensional subspace. The second moment of the partial period cross-correlation can be expressed in terms of the cardinalities of certain fourfold intersections of these hyperplanes, as follows.

Lemma 4.2 *If* $\mathbf{S}$ *and* $\mathbf{T}$ *are geometric sequences, then*

$$E[\Theta_{\mathbf{S},\mathbf{T}}(\tau, k, D)^2] =$$
$$\frac{1}{q^n - 1}\sum_{i,j=0}^{D-1}\left(\sum_{\substack{s,t,u,v\\ \in GF(q)}} N_{i,j,\tau}(s,t,u,v)F(s)G'(t)F(u)G'(v) - 1\right) \qquad (1)$$

where $G'(x) = G(x^{p^e})$ *and* $N_{i,j,\tau}(s,t,u,v) = |H^s_{\alpha^i} \cap H^t_{\alpha^{i+\tau}} \cap H^u_{\alpha^j} \cap H^v_{\alpha^{j+\tau}}|$.

The results of [7] give us the number of times that $N_{i,j,\tau}(s,t,u,v)$ is nonzero and the values of $N_{i,j,\tau}(s,t,u,v)$ for each i, j, and τ. There are three cases depending on whether $\alpha^\tau \in GF(q)$, $\alpha^\tau \in GF(q^2) - GF(q)$, or $\alpha^\tau \in GF(q^n) - GF(q^2)$. For each of these cases we count the number of times each value of $N_{i,j,\tau}(s,t,u,v)$ occurs. This allows us to decompose the sum in equation (1) according to the values of $N_{i,j,\tau}(s,t,u,v)$. The contribution from the instances when the hyperplanes are in general position is essentially cancelled by subtracting the square of the expectation in the final calculation of the variance. Our estimates are derived by bounding the number of times the hyperplanes are not in general position.

We next give conditions in terms of i, j, and τ under which the various possible values of $N_{i,j,\tau}(s,t,u,v)$ occur. For the remainder of this section we fix i, j, τ, with $0 \leq i, j, \tau \leq q^n - 2$, and let $A = \alpha^i$, $B = \alpha^j$, $C = \alpha^\tau \in GF(q^n)$. For any $s, t, u, v \in GF(q^n)$, define the affine linear subspace

$$Z(s,t,u,v) = H_A^s \cap H_{AC}^t \cap H_B^u \cap H_{BC}^v.$$

Then $N_{i,j,\tau}(s,t,u,v) = |Z(s,t,u,v)|$. There are five possible values for $N_{i,j,\tau}(s,t,u,v)$: q^{n-4}, q^{n-3}, q^{n-2}, q^{n-1}, and 0, depending on the dimension of $Z(s,t,u,v)$. Let r be the dimension of the $GF(q)$-vector space spanned by the elements $\{A, AC, B, BC\}$ when we think of $GF(q^n)$ as a vector space over $GF(q)$.

Proposition 4.3 *(Klapper and Goresky [7]) If there exist $a, b, c, d \in GF(q)$ such that*

$$aA + bAC + cB + dBC = 0, \tag{2}$$

and $as + bt + cu + dv \neq 0$, then $Z(s,t,u,v) = \emptyset$ so $N_{i,j,\tau}(s,t,u,v) = 0$. Otherwise, $\dim Z(s,t,u,v) = n - r$ so $N_{i,j,\tau}(s,t,u,v) = q^{n-r}$. There are q^r values of (s,t,u,v) such that $N_{i,j,\tau}(s,t,u,v) = q^{n-r}$, and these are exactly the points in the image of the function

$$L_{i,j,\tau}(x) = (Tr_q^{q^n}(Ax), Tr_q^{q^n}(ACx), Tr_q^{q^n}(Bx), Tr_q^{q^n}(BCx)).$$

There is an action of the general linear group over $GF(q)$ of rank two, $G = GL_2(GF(q))$, on $GF(q^n)$. Recall that this group is the multiplicative group of two by two matrices with nonzero determinant and with entries in $GF(q)$, acting by fractional linear transformations. That is, the matrix

$$M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

acts on the element $C \in GF(q)$ by

$$C \mapsto \frac{aC + b}{cC + d} = M(C).$$

If $M, N \in GL_2(GF(q))$, then $(MN)(C) = M(N(C))$. Recall that when a group G acts on a set W, the *G-orbit* of an element $x \in W$ is the set $\text{orbit}(x) = \{M(x) : M \in G\}$.

If $C \notin GF(q)$ then an equation of linear dependence (2),

$$aAC + bA + cBC + dB = 0$$

may be interpreted as an equation $B/A = -M(C)$, where $M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$. If $ad - bc \neq 0$, then $M \in Gl_2(GF(q))$, in other words, $B/A \in \text{orbit}(C)$. If $ad - bc = 0$ then $B/A \in GF(q)$.

Theorem 4.4 *(Klapper and Goresky [7]) Let r be as in Proposition 4.3.*

1. *Suppose $C = \alpha^\tau \in GF(q)$.*
   (a) *If $B/A = \alpha^{j-i} \in GF(q)$, then $r = 1$ (so $N_{i,j,\tau}(0,0,0,0) = q^{n-1}$);*
   (b) *otherwise $r = 2$ (so $N_{i,j,\tau}(0,0,0,0) = q^{n-2}$).*

2. *Suppose* $C = \alpha^\tau \in GF(q^2) - GF(q)$.
   (a) *If* $B/A = \alpha^{j-i} \in GF(q)$, *then* $r = 2$ *(so* $N_{i,j,\tau}(0,0,0,0) = q^{n-2}$*);*
   (b) *if* $\alpha^{j-i} \in \mathrm{orbit}(\alpha^\tau)$, *then* $r = 2$ *(so* $N_{i,j,\tau}(0,0,0,0) = q^{n-2}$*);*
   (c) *otherwise* $r = 4$ *(so* $N_{i,j,\tau}(0,0,0,0) = q^{n-4}$*).*
3. *Suppose* $C = \alpha^\tau \in GF(q^n) - GF(q^2)$.
   (a) *If* $B/A = \alpha^{j-i} \in GF(q)$, *then* $r = 2$ *(so* $N_{i,j,\tau}(0,0,0,0) = q^{n-2}$*);*
   (b) *if* $B/A = \alpha^{j-i} \in \mathrm{orbit}(\alpha^\tau)$, *then* $r = 3$ *(so* $N_{i,j,\tau}(0,0,0,0) = q^{n-3}$*);*
   (c) *otherwise* $r = 4$ *(so* $N_{i,j,\tau}(0,0,0,0) = q^{n-4}$*).*

In computing the variance of the partial period correlation of geometric sequences, equation (1), we break down our analysis depending upon whether α^τ is in $GF(q)$, $GF(q^2) - GF(q)$, or $GF(q^n) - GF(q^2)$. This breakdown corresponds to the three cases of Theorem 4.4. We remark that, if n is odd, then $GF(q^2)$ is not a subfield of $GF(q^n)$, so the middle case will not occur. In order to use parts 2(b) and 3(b) of Theorem 4.4 it is necessary to determine, for given $0 \le i, j < D$, whether $\alpha^{i-j} \in \mathrm{orbit}(\alpha^\tau)$.

The most difficult case occurs when $\alpha^\tau \in GF(q^n) - GF(q^2)$. This corresponds to Theorem 4.4 part (3). We estimate the second moment by estimating the number of i and j for which each of the cases 3(a), 3(b), and 3(c) occurs. Case 3(b) gives rise to the dominant term in our estimate, so we need to understand the distribution of points in a G-orbit. In general, the G-orbit is not uniformly distributed in $GF(q^n)$, so for a fixed i, the number of j in a window with $\dim\{\alpha^{i+\tau}, \alpha^i, \alpha^{j+\tau}, \alpha^j\} = 3$ is not proportional to the size of the window. We settle here for a cruder estimate, based on the structure of the group G. Bounds on the size of the intersection a G-orbit were found by Klapper and Goresky [7].

Proposition 4.5 *The intersection of the orbit of* x *with a window of size* D, D *arbitrary, has cardinality at least* $\lfloor D/\nu \rfloor (q^2 + q)$ *and at most* $\lceil D/\nu \rceil (q^2 + q)$.

This estimate can then be used to complete the estimates in Theorem 4.1. These (perhaps crude) estimates arise by assuming all terms of the form $F(s)G'(t)F(u)G'(v)$ equal one, the worst case. In general are plus or minus one, so there may be some cancellation. A further reduction in the estimate of the second moment can be proved when f or g is balanced. In this case, enough cancellation occurs in these inner sums over s, t, u, and v that we can reduce the estimate of the variance by a factor of two.

The remaining cases ($\alpha \in GF(q)$ and $\alpha \in GF(q^2) - GF(q)$) are similar, and in fact simpler.

References

[1] A. H. CHAN AND R. GAMES, On the linear span of binary sequences from finite geometries, q odd, *in Proceedings of Crypto 1986,* pp. 405-417, Santa Barbara.

[2] A. H. CHAN, M. GORESKY, AND A. KLAPPER, Cross-correlations of linearly and quadratically related geometric sequences and GMW sequences, to appear, *Discrete Applied Mathematics.*

[3] S. Golomb, *Shift Register Sequences*, Aegean Park Press, Laguna Hills, CA, 1982.

[4] B. Gordon, W. H. Mills, and L. R. Welch, Some new difference sets, *Canad. J. Math* **14** (1962) pp. 614-625.

[5] T. Hoholdt, H. E. Jensen, and J. Justesen, Aperiodic correlations and the merit factor of a class of binary sequences, *IEEE Trans. on Inf. Th.* **31** (1985), pp. 549-552.

[6] A. Klapper, A.H. Chan, and M. Goresky, Cascaded GMW sequences, to appear in *IEEE Trans. Inf. Thy.*.

[7] A. Klapper and M. Goresky, Partial period autocorrelations of geometric sequences, submitted to *IEEE Trans. Inform. Theory.*

[8] P. V. Kumar, The partial-period correlation moments of arbitrary binary sequences, *GLOBECOM '85*, IEEE Global Telecommunications Conference – Conference Record, IEEE Publications, N.Y., New York (1985).

[9] R. Lidl and H. Niederreiter, *Finite Fields, Encyclopedia of Mathematics vol. 20*, Cambridge University Press, Cambridge, 1983.

[10] R. McEliece, *Finite Fields for Computer Scientists and Engineers,* Kluwer Academic Publishers, Boston, 1987.

[11] O. Rothaus, On bent functions, *J. of Combinatorial Theory, Series A* **20** (1976), pp. 300-305.

[12] R. A. Rueppel *Analysis and Design of Stream Ciphers*, Springer Verlag, New York, 1986.

[13] R. A. Scholtz and L. R. Welch, GMW sequences, *IEEE Trans. on Inf. Theory* **IT-30**, pp. 548-553.

[14] M. Simon, J. Omura, R. Scholtz, and B. Levitt, *Spread-Spectrum Communications, Vol. 1*, Computer Science Press, 1985.

PARTIAL EXPANSION PROPERTIES OF LINEAR BLOCK CODES

ALI S. KHAYRALLAH

ELECTRICAL ENGINEERING DEPARTMENT
UNIVERSITY OF DELAWARE
NEWARK, DE 19716

ABSTRACT

In previous work [2] [3], we introduced the expansion factor of a linear block code. Here we investigate the ℓ-expansion factor $\Delta^{(\ell)}(C)$, which is a measure of how much the code preserves and expands the distance relations among prefixes of length ℓ of input words of length $k \geq \ell$. One can view $\Delta^{(\ell)}(C)$ as a measure of the continuity of the code. We relate $\Delta^{(\ell)}(C)$ to n, k, d, the total weight, and the weight distribution. We find $\Delta^{(\ell)}(C)$ for the Hamming code, the Golay code, and BCH codes.

INTRODUCTION

Consider the problem of error-control coding over the q-ary symmetric channel using linear block codes. This paper is concerned with the relation between input and codeword distance in a code. Namely, it introduces the partial expansion factor, or ℓ-expansion factor, which is a measure of how much the code preserves and expands the distance relations among prefixes of length ℓ of the input words of length $k \geq \ell$. The relevant distance is the familiar Hamming distance $d(\mathbf{u},\mathbf{v})$. We also use the Hamming weight $w(\mathbf{u}) = d(\mathbf{u},\mathbf{0})$. This paper generalizes our earlier work in [2] [3], which dealt with the case $k = \ell$. The subject is related to that of [6], which is concerned with convolutional codes.

An $(n,k;d)$ minimum distance code has input length k, output length n, minimum distance d, and an alphabet A of size q. We assume A to be the field GF(q) with the appropriate addition and multiplication operators. The encoder is described by a $(k \times n)$ generator matrix $\mathbf{G}$, which maps an input $\mathbf{x} \in A^k$ into a codeword $\mathbf{y} = \mathbf{x}\,\mathbf{G} \in A^n$. The codebook C is defined as the row-space of $\mathbf{G}$

$$C = C(\mathbf{G}) = \{\mathbf{y}:\ \mathbf{y} = \mathbf{x}\,\mathbf{G},\ \mathbf{x} \in A^k\}$$

Also, recall that the minimum distance d can be written as

$$d \overset{\Delta}{=} \min_{\substack{\mathbf{x} \in A^k \\ \mathbf{x} \neq \mathbf{0}}} w(\mathbf{x}\ \mathbf{G})$$

At the output of the channel, the decoder can be described in two stages. The first stage is a decision rule $\mathbf{D}$ that maps the channel output $\mathbf{z} \in A^n$ into a codeword estimate $\hat{\mathbf{y}} = \mathbf{D}(\mathbf{z}) \in C$. In this paper, $\mathbf{D}$ is a bounded distance decoding rule, which we discuss later. The second stage of the decoder is an "uncoder"[1], which simply maps $\hat{\mathbf{y}}$ into its corresponding input $\hat{\mathbf{x}}$. It can be conveniently described by an $(n \times k)$ matrix $\mathbf{G}^{-1}$ such that $\mathbf{G}\,\mathbf{G}^{-1} = \mathbf{I}$. Then $\hat{\mathbf{x}} = \hat{\mathbf{y}}\,\mathbf{G}^{-1} \in A^k$. The error-control code components are shown in Figure 1.

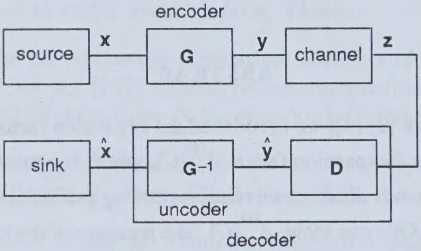


Figure 1. Error-control code components

Let $\Gamma(C)$ denote the set of all generator matrices that produce C

$$\Gamma(C) = \{\mathbf{G}:\ C(\mathbf{G}) = C\}$$

The minimum distance d is a property of the code C, in the sense that it does not depend on the generator matrix $\mathbf{G}$. In the remainder, we will refer to the codebook C and the decision rule $\mathbf{D}$ as the code. A matrix $\mathbf{G}$ in $\Gamma(C)$ (along with the uncoder matrix $\mathbf{G}^{-1}$) is viewed as a specific realization of the code. We are concerned with the choice of $\mathbf{G}$. Our criterion is the expansion factor of $\mathbf{G}$, which we introduce next.

For an input word $\mathbf{x} = (x_0 \ldots x_{k-1})$, we denote the prefix of length ℓ as $\mathbf{x}^{(\ell)} = (x_0 \ldots x_{\ell-1})$, $\ell = 1, \ldots, k$. In many situations, the input letters x_i are of varying importance to the user, and it pays to try to protect some of them more than others. Here, we assume that the user wishes to minimize the impact of channel errors on $\mathbf{x}^{(\ell)}$ for some ℓ. For example, $\mathbf{x}^{(\ell)}$ may consist of the most significant bits in a natural binary representation of integers, or the header portion of a packet in a communication network, etc.

In the coding scenario of Figure 1, $\mathbf{x}$ is mapped into $\mathbf{y} = \mathbf{x}\,\mathbf{G}$, which is sent over the channel. Suppose a channel error event occurs, that is $\mathbf{z} \neq \mathbf{y}$, and that it leads to a code error event, that is $\hat{\mathbf{y}} = \mathbf{D}(\mathbf{z}) \neq \mathbf{y}$. The corresponding input word is $\hat{\mathbf{x}} = \hat{\mathbf{y}}\,\mathbf{G}^{-1} \neq \mathbf{x}$. We refer to $d(\mathbf{y},\mathbf{z})$ and $d(\mathbf{y},\hat{\mathbf{y}})$ as the channel error and code error, respectively. The user would like to minimize the user error $d(\mathbf{x}^{(\ell)},\hat{\mathbf{x}}^{(\ell)})$. This is achieved by some $\mathbf{G}$ in $\Gamma(C)$ (and its $\mathbf{G}^{-1}$). (If $\ell < k$, one might even get $\mathbf{x}^{(\ell)} = \hat{\mathbf{x}}^{(\ell)}$, concealing the code error completely.) Clearly, we need to judge $\mathbf{G}$ by how it works for all possible inputs and code errors. Now every pair $(\mathbf{x},\hat{\mathbf{x}})$, regardless of how large $d(\mathbf{x}^{(\ell)},\hat{\mathbf{x}}^{(\ell)})$ is, corresponds to some pair $(\mathbf{y},\hat{\mathbf{y}})$. Since large user errors cannot be eliminated, we try to do the next best thing, which is to link them with unlikely code error events. The probability of an error event $\hat{\mathbf{y}} \neq \mathbf{y}$ decreases (typically sharply) with $d(\mathbf{y},\hat{\mathbf{y}})$. (We discuss this in some detail later.) So we look at how well $\mathbf{G}$ matches code error $d(\mathbf{y},\hat{\mathbf{y}})$ to user error $d(\mathbf{x}^{(\ell)},\hat{\mathbf{x}}^{(\ell)})$. This way, likely error events (small $d(\mathbf{y},\hat{\mathbf{y}})$) result in small $d(\mathbf{x}^{(\ell)},\hat{\mathbf{x}}^{(\ell)})$, while large $d(\mathbf{x}^{(\ell)},\hat{\mathbf{x}}^{(\ell)})$ are caused by unlikely error events (large $d(\mathbf{y},\hat{\mathbf{y}})$). Our definition follows. Given the generator matrix $\mathbf{G}$ of an $(n,k;d)$ code, the ℓ-expansion factor $\delta^{(\ell)}$ is defined as

---

[1] This name was coined by Chris Heegard.

$$\delta^{(\ell)} \triangleq \min_{\mathbf{x}:\ \mathbf{x}^{(\ell)}\neq 0} w(\mathbf{x}\,\mathbf{G}) - w(\mathbf{x}^{(\ell)})$$

That is, for any $\mathbf{x}$ such that $\mathbf{x}^{(\ell)} \neq \mathbf{0}$, $w(\mathbf{x}\,\mathbf{G}) \geq \max\,(d\,,\, w(\mathbf{x}^{(\ell)}) + \delta^{(\ell)})$. The code's linearity allows us to express $\delta^{(\ell)}$ in terms of weight instead of distance.

We now define the partial expansion factor or ℓ-expansion factor $\Delta^{(\ell)}(C)$ of the code as

$$\Delta^{(\ell)}(C) \triangleq \max_{\mathbf{G}\in\Gamma(C)} \delta^{(\ell)}(\mathbf{G})$$

We also pick $\mathbf{G}^{(\ell)}$ in $\Gamma(C)$ with $\delta^{(\ell)}(\mathbf{G}^{(\ell)}) = \Delta^{(\ell)}(C)$ and call it the ℓ-expansion matrix of C. It is easy to see that if $\mathbf{G}$ and $\mathbf{G}' \in \Gamma(C)$, then there is a unique $(k \times k)$ matrix $\mathbf{F}$ such that $\mathbf{G}' = \mathbf{F}\,\mathbf{G}$. We now argue that one can always choose to use a $\mathbf{G}^{(\ell)}$, basically for free. Suppose that the components $\mathbf{G}$, $\mathbf{D}$, and $\mathbf{G}^{-1}$ have already been chosen by the user. Also suppose that $\delta^{(\ell)}(\mathbf{G}) < \Delta^{(\ell)}(C)$. Now let $\mathbf{F}$ satisfy $\mathbf{G}^{(\ell)} = \mathbf{F}\,\mathbf{G}$. Then all we need is the addition of two new components, $\mathbf{F}$ as a first stage in the encoder, and $\mathbf{F}^{-1}$ as a last stage in the uncoder (since $(\mathbf{G}^{(\ell)})^{-1} = \mathbf{G}^{-1}\,\mathbf{F}^{-1}$). Compared to the cost of the decision rule $\mathbf{D}$, it is reasonable to assume that of $\mathbf{F}$ and $\mathbf{F}^{-1}$ to be negligible.

EXAMPLE

Consider the (7,4;3) Hamming code [5]. Its generator polynomial is given by $g(X) = 1 + X + X^2 + X^4$. Two matrices for this code are

$$\mathbf{G} = \begin{bmatrix} 1&1&0&1&0&0&0\\ 0&1&1&0&1&0&0\\ 0&0&1&1&0&1&0\\ 0&0&0&1&1&0&1 \end{bmatrix} \qquad \mathbf{G}' = \begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&1&0\\ 1&0&0&1 \end{bmatrix} \mathbf{G} = \begin{bmatrix} 1&1&0&1&0&0&0\\ 0&1&1&0&1&0&0\\ 0&0&1&1&0&1&0\\ 1&1&0&0&1&0&1 \end{bmatrix}$$

where $\mathbf{G}$ is the direct matrix representation of $g(X)$. Note that for $\mathbf{x} = (1\ 1\ 1\ 1)$, $w(\mathbf{x}^{(4)}) = 4$, $\mathbf{y} = \mathbf{x}\,\mathbf{G}' = (0\ 1\ 0\ 0\ 0\ 1\ 1)$, and $w(\mathbf{y}) = 3$. So from the viewpoint of that particular input, encoding with $\mathbf{G}'$ is detrimental to its error protection. It is easy to check that $\delta^{(4)}(\mathbf{G}') = -1$ while $\delta^{(4)}(\mathbf{G}) = 0$, so $w(\mathbf{x}\,\mathbf{G}) \geq w(\mathbf{x}^{(4)})$ for all $\mathbf{x}$. It also turns out $\Delta^{(4)}(C) = 0$, and $\mathbf{G}^{(4)} = \mathbf{G}$. The Hamming code is revisited in a later section, where the rest of its ℓ-expansion factors and matrices are given, as well as those of other codes.

EXPANSION FACTOR AND BOUNDED DISTANCE DECODER

Consider a code C whose decision rule $\mathbf{D}$ is a bounded distance decoder with radius ρ. The code is realized by a generator matrix $\mathbf{G}$ with ℓ-expansion factor $\delta^{(\ell)}(\mathbf{G}) = \delta$. Refer to Figure 1 for $\mathbf{x}$, $\mathbf{y}$, $\mathbf{z}$, $\hat{\mathbf{y}}$, and $\hat{\mathbf{x}}$. The decision rule $\mathbf{D}$, from A^n to C, is given by

$$\mathbf{D}(\mathbf{z}) = \hat{\mathbf{y}} \ \text{ if } \ d(\mathbf{z},\hat{\mathbf{y}}) \leq \rho$$

This is shown in Figure 2.

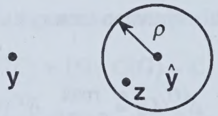


Figure 2. Error event in a bounded distance decoder

It follows from the definition of $\delta^{(\ell)}(\mathbf{G})$ that

$$d(\mathbf{x}^{(\ell)},\hat{\mathbf{x}}^{(\ell)}) \leq d(\mathbf{y},\hat{\mathbf{y}}) - \delta \leq d(\mathbf{y},\mathbf{z}) + \rho - \delta \tag{1}$$

We can see that knowledge of δ provides an upper bound to the user error in terms of the code error, and the channel error. The probability of decoding $\hat{\mathbf{y}}$ given $\mathbf{y}$ depends only on $d(\mathbf{y},\hat{\mathbf{y}})$, that is

$$p(\hat{\mathbf{y}}|\mathbf{y}) \triangleq \Pi_{d(\mathbf{y},\hat{\mathbf{y}})}$$

The exact formula is given in [4]. In [1], it is shown that Π_u is bounded by

$$\binom{d}{\rho}\, p_{u-\rho} \leq \Pi_u \leq s_\rho\, p_{u-\rho}$$

where $p_v = (\varepsilon/(q-1))^v\,(1-\varepsilon)^{n-v}$ and $s_\rho = \sum_{0\leq j\leq\rho} \binom{n}{j}\,(q-1)^j$. While we don't know of a proof that Π_u decreases with u, one can see that both bounds to Π_u decrease with

u, which confirms our earlier remark that distant error events tend to have low probability. From the user's viewpoint, note that roughly speaking, $p(\hat{\mathbf{x}}|\mathbf{x}) \cong \Pi_{d(\mathbf{x}^{(\ell)},\hat{\mathbf{x}}^{(\ell)})+\delta}$. So a bigger δ means that large user errors have a lower probability. This is a good reason for using an expansion matrix $\mathbf{G}^{(\ell)}$.

PROPERTIES OF ℓ-EXPANSION FACTORS

In this section and the next, we present a number of results on ℓ-expansion factors.

Proposition 1

The partial expansion factors $\delta^{(\ell)}$, $1 \le \ell \le k$, of the generator matrix $\mathbf{G}$ of an $(n,k;d)$ code satisfy

$$\delta^{(\ell)} \ge d - \ell \qquad 1 \le \ell \le k-1 \qquad (2)$$

$$\delta^{(\ell)} \ge \delta^{(\ell+1)} \qquad 1 \le \ell \le k-1 \qquad (3)$$

Proof

In the worst case, $\mathbf{x}$ with $w(\mathbf{x}^{(\ell)}) = \ell$ is mapped into $\mathbf{y}$ with $w(\mathbf{y}) = d$. This shows the inequality (2). Now consider $\mathbf{x}$ with $w(\mathbf{y}) - w(\mathbf{x}^{(\ell)}) = \delta^{(\ell)}$. Since $w(\mathbf{x}^{(\ell)}) \le w(\mathbf{x}^{(\ell+1)})$, we have $w(\mathbf{y}) - w(\mathbf{x}^{(\ell+1)}) \le \delta^{(\ell)}$. In addition $w(\mathbf{y}) - w(\mathbf{x}^{(\ell+1)}) \ge \delta^{(\ell+1)}$, so $\delta^{(\ell)} \ge \delta^{(\ell+1)}$, which shows inequality (3).

We say that a generator matrix $\mathbf{G}$ is ℓ-systematic if there exist j_i, $0 \le i \le \ell-1$, such that for any codeword $\mathbf{y} = \mathbf{x}\,\mathbf{G}$, $y_{j_i} = x_i$. It is easy to show that $\Gamma(C)$ contains an ℓ-systematic matrix for all ℓ.[2] Also, if $\mathbf{G}$ is ℓ-systematic, then $\delta^{(\ell)}(\mathbf{G}) \ge 0$. This, combined with inequality (3) and the definition of $\Delta^{(\ell)}(C)$, shows the next result.

Proposition 2

The partial expansion factors of an $(n,k;d)$ code C satisfy

$$\Delta^{(1)}(C) \ge \Delta^{(2)}(C) \ge \ldots \ge \Delta^{(k)}(C) \ge 0$$

Note that there is a tradeoff between the number ℓ of letters being protected, and the amount of protection $\Delta^{(\ell)}(C)$.

---

[2] We do not require $y_i = x_i$, $0 \le i \le \ell-1$, since a $\mathbf{G}$ satisfying this condition may not belong to $\Gamma(C)$, strictly speaking.

WEIGHT DISTRIBUTION

The familiar weight density (or coefficients) a_i denotes the number of codewords of weight i, $0 \le i \le n$. The weight distribution b_i is defined as

$$b_i \overset{\Delta}{=} \sum_{j=0}^{i} a_j$$

In the set A^k, the number of words $\mathbf{x}$ of weight $w(\mathbf{x}^{(\ell)}) = i$ is given by

$$q^{k-\ell} \binom{\ell}{i} (q-1)^i$$

for $0 \le i \le \ell$. Now suppose there exists a code C and a $\mathbf{G}^{(\ell)}$ such that $\Delta^{(\ell)}(C) = \delta$ is achieved with the smallest possible codeword weights. That is, all $\mathbf{x} \ne \mathbf{0}$ with $w(\mathbf{x}^{(\ell)}) \le \max(0, d-\delta)$ map into $\mathbf{y}$ with $w(\mathbf{y}) = d$, and all $\mathbf{x}$ with $w(\mathbf{x}^{(\ell)}) > \max(0, d-\delta)$ map into $\mathbf{y}$ with $w(\mathbf{y}) = w(\mathbf{x}^{(\ell)}) + \delta$. It is easy to show that its weight distribution $b_i^{(\ell,\delta)}$ is given by

$$b_i^{(\ell,\delta)} \overset{\Delta}{=} \begin{cases} 1 & 0 \le i \le d-1 \\ q^{k-\ell} + q^{k-\ell} \sum_{j=1}^{i-\delta} \binom{\ell}{j} (q-1)^j & d \le i \le \ell + \delta \\ q^k & \ell + \delta + 1 \le i \le n \end{cases}$$

Note that the above code may not exist (and often doesn't) for given n, ℓ, d, and δ. It is just a way to generate $b_i^{(\ell,\delta)}$.

Proposition 3

The weight distribution b_i of an $(n,k;d)$ code with ℓ-expansion factor $\Delta^{(\ell)}(C) = \delta$ satisfies

$$b_i \le b_i^{(\ell,\delta)} \qquad \text{all } i \tag{4}$$

Proof

For $i \le d-1$, $b_i = b_i^{(\ell,\delta)} = 1$, since the code has minimum distance d. Now suppose $b_i > b_i^{(\ell,\delta)}$ for some $i \ge d$. By construction of the above hypothetical code, $b_i^{(\ell,\delta)}$ is equal to the number of input words of weight $w(\mathbf{x}^{(\ell)}) \le i - \delta$. Their remains $b_i - b_i^{(\ell,\delta)}$ codewords of weight no greater than i, and there are no input words left of weight $w(\mathbf{x}^{(\ell)}) \le i - \delta$. Hence no $\mathbf{G}$ can achieve $\delta^{(\ell)}(\mathbf{G}) = \delta$. This contradicts $\Delta^{(\ell)}(C) = \delta$.

It is easy to check that $b_i^{(\ell,\delta)}$ decreases with δ for a fixed ℓ. In light of Proposition 3, a code with a large $\Delta^{(\ell)}(C) = \delta$ has its weight distribution b_i "pushed" to higher weights i, which means that the code has many large weight codewords. Intuitively, one would suspect that such a code has a low rate. Indeed, this is true, and it is made clearer in the next result.

TOTAL WEIGHT

We define the total weight W of a code as the sum of all codeword weights. It can be written as

$$W \overset{\Delta}{=} \sum_{i=0}^{n} a_i \, i$$

$$= \sum_{i=1}^{n} (b_i - b_i) \, i \tag{5}$$

The total weight W can be computed directly from n, k, and q [7]

$$W = n\,(q-1)\,q^{k-1}$$

Note that for fixed k and q, the rate k/n decreases with W.

For the hypothetical code with weight distribution $b_i^{(\ell,\delta)}$, the total weight $W^{(\ell,\delta)}$ can be written as

$$W^{(\ell,\delta)} \overset{\Delta}{=} \sum_{i=d}^{\ell+\delta} (b_i^{(\ell,\delta)} - b_{i-1}^{(\ell,\delta)}) \, i \tag{6}$$

Proposition 4

The total weight W of an $(n,k;d)$ code C with $\Delta^{(\ell)}(C) = \delta$ satisfies

$$W \geq W^{(\ell,\delta)} \tag{7}$$

Proof

The proof follows directly from (5) and (6), and the inequalities (4).

It is easy to check that $W^{(\ell,\delta)}$ increases with δ. Thus a code with a large expansion factor must have a large total weight. In turn, this means that the rate k/n has to be low.

Proposition 3 yields an upper bound $\Delta_b^{(\ell)}$ to $\Delta^{(\ell)}(C)$ in terms of the weight distribution b_i. That is, $\Delta_b^{(\ell)}$ is the smallest value of δ such that (4) holds. Similarly, Proposition 4 yields a looser upper bound $\Delta_W^{(\ell)}$ to $\Delta^{(\ell)}(C)$ in terms of the total weight W. That is, $\Delta_W^{(\ell)}$ is the smallest value of δ such that (7) holds. Note that the latter bound is found without knowledge of the weight distribution of the code, which is sometimes unavailable.

EXAMPLES

We find the expansion factors and their bounds for several familiar codes. We also give a few of the corresponding expansion matrices $\mathbf{G}^{(\ell)}$. It is interesting to note that $\mathbf{G}^{(\ell)}$ changes with ℓ.

First consider the (7,4;3) Hamming code, whose generator polynomial $g(X)$ and its corresponding generator matrix $\mathbf{G}$ were given earlier. The expansion factors and their bounds are listed in Table 1.

ℓ	$\Delta^{(\ell)}(C)$	$\Delta_b^{(\ell)}$	$\Delta_W^{(\ell)}$
1	2	3	3
2	2	2	2
3	1	1	2
4	0	1	1

Table 1. Expansion factors and bounds for the (7,4,3) Hamming code.

We already mentioned that $\mathbf{G}^{(4)} = \mathbf{G}$. We also find that $\mathbf{G}^{(1)} = \mathbf{G}$, while

$$\mathbf{G}^{(2)} = \mathbf{G}^{(3)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix} \mathbf{G} = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}$$

The (15,5;7) BCH code [5] has polynomial $g(X) = 1 + X + X^2 + X^4 + X^5 + X^8 + X^{10}$, which yields

$$\mathbf{G} = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}$$

The expansion factors and their bounds are listed in Table 2. Here we find $\mathbf{G}^{(1)} = \mathbf{G}$,

$$\mathbf{G}^{(2)} = \mathbf{G}^{(3)} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{G} = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$$

$$\mathbf{G}^{(4)} = \begin{bmatrix} 1&0&0&0&0 \\ 0&1&0&0&0 \\ 0&0&1&0&0 \\ 0&0&0&1&0 \\ 1&0&0&0&1 \end{bmatrix} \mathbf{G} = \begin{bmatrix} 1&1&1&0&1&1&0&0&1&0&1&0&0&0&0 \\ 0&1&1&1&0&1&1&0&0&1&0&1&0&0&0 \\ 0&0&1&1&1&0&1&1&0&0&1&0&1&0&0 \\ 0&0&0&1&1&1&0&1&1&0&0&1&0&1&0 \\ 1&1&1&0&0&0&1&0&0&1&1&0&1&0&1 \end{bmatrix}$$

$$\mathbf{G}^{(5)} = \begin{bmatrix} 1&0&0&0&0 \\ 0&1&0&0&0 \\ 0&0&1&0&0 \\ 0&0&0&1&0 \\ 0&1&1&0&1 \end{bmatrix} \mathbf{G} = \begin{bmatrix} 1&1&1&0&1&1&0&0&1&0&1&0&0&0&0 \\ 0&1&1&1&0&1&1&0&0&1&0&1&0&0&0 \\ 0&0&1&1&1&0&1&1&0&0&1&0&1&0&0 \\ 0&0&0&1&1&1&0&1&1&0&0&1&0&1&0 \\ 0&1&0&0&0&0&1&1&1&0&1&1&0&0&1 \end{bmatrix}$$

For the (15,7;5) BCH code [5] with polynomial $g(X) = 1 + X^4 + X^6 + X^7 + X^8$, the results are listed in Table 3. We do not list the ℓ-expansion matrices for this and the last two codes for lack of space.

ℓ	$\Delta^{(\ell)}(C)$	$\Delta_b^{(\ell)}$	$\Delta_W^{(\ell)}$
1	6	7	7
2	6	6	6
3	5	5	6
4	4	4	5
5	4	4	5

Table 2. Expansion factors and bounds for the (15,5;7) BCH code.

ℓ	$\Delta^{(\ell)}(C)$	$\Delta_b^{(\ell)}$	$\Delta_W^{(\ell)}$
1	4	5	5
2	4	5	5
3	4	4	5
4	3	4	5
5	2	4	5
6	2	3	4
7	2	3	4

Table 3. Expansion factors and bounds for the (15,7,5) BCH code.

For the (15,11;3) BCH code [5] with polynomial $g(X) = 1 + X + X^4$, the results are listed in Table 4. Finally, for the (23,12;7) Golay code [5] with polynomial $g(X) = 1 + X^2 + X^4 + X^5 + X^6 + X^{10} + X^{11}$, the results are listed in Table 5.

ℓ	$\Delta^{(\ell)}(C)$	$\Delta_b^{(\ell)}$	$\Delta_W^{(\ell)}$
1	2	3	3
2	1	3	3
3	1	3	3
4	1	3	3
5	0	3	3
6	0	2	3
7	0	2	3
8	0	2	3
9	0	2	3
10	0	1	2
11	0	1	2

Table 4. Expansion factors and bounds for the (15,11;3) BCH code

ℓ	$\Delta^{(\ell)}(C)$	$\Delta_b^{(\ell)}$	$\Delta_W^{(\ell)}$
1	6	7	7
2	5	7	7
3	4	7	7
4	3	7	7
5	3	6	7
6	2	6	7
7	2	6	7
8	2	5	7
9	1	5	7
10	1	4	6
11	1	4	6
12	0	4	5

Table 5. Expansion factors and bounds for the (23,12;7) Golay code

CONCLUSIONS

We investigated the partial expansion factors of linear block codes. One can view them as measures of the continuity of the code. We related the partial expansion factors to the structural properties of the code, namely its weight coefficients and total weight. We computed the partial expansion factors for a number of well known codes.

A partial expansion factor is not an alternative to minimum distance in evaluating a code, but it can be a very useful second figure of merit. It is relevant whenever input distance is meaningful. In many cases, only part of the input may have meaningful distance. Our treatment gives the user the flexibility to pick the appropriate value of ℓ.

Preliminary results indicate that ℓ-expansion matrices yield a small per letter error probability $\sum_{i=0}^{\ell-1} P(x_i \neq \hat{x}_i) / \ell$. This tends to agree with how ℓ-expansion matrices match large user errors with unlikely code error events.

BIBLIOGRAPHY

[1] A. S. Khayrallah, "Per-letter error probability of expansion channel codes," *Proceedings of the Allerton Conference on Communication, Control, and Computing* (1992).

[2] A. S. Khayrallah, "The expansion factor of error control codes," *Abstracts of the IEEE International Symposium on Information Theory* (1993).

[3] A. S. Khayrallah, "The expansion factor of error control codes," *Submitted to IEEE Transactions on Information Theory* (1993).

[4] R. E. Blahut, Theory and practice of error control codes, Addison-Wesley (1984).

[5] S. Lin and D. J. Costello, Error control coding: Fundamentals and applications, Prentice-Hall (1983).

[6] D. G. Mills and D. J. Costello, "Using a modified transfer function to determine the unequal error protection capabilities of convolutional codes," *Abstracts of the IEEE International Symposium on Information Theory* (1993).

[7] W. W. Peterson and E. R. Weldon, Error-correcting codes, second edition, MIT Press (1972).

Bit-Level Parallel Decoding of Reed-Solomon Codes

Siddhartha Ray-Chaudhuri[1] and Agnes H. Chan[2]

Introduction Reed-Solomon (RS) codes are a class of M-ary cyclic codes ($M > 2$), which are often used to correct bit-level burst errors [1]. RS codes are commonly decoded by M-ary bounded-distance symbol-level (BDSL) decoders. If the uncoded symbol error rate is large, the performance of BDSL decoder deteriorates considerably. This paper studies a bit-level parallel decoding algorithm for RS codes, called *split decoding,* which outperforms BDSL decoding for several classes of 2^m-ary RS codes at high channel error rates.

Split Decoding with Coordinate Codes Let $(c_{i,0}, c_{i,1}, c_{i,2}, \ldots, c_{i,m-1})$ be a binary expansion of symbol c_i in codeword $\mathbf{c} = (c_0, c_1, c_2, \ldots, c_{N-1})$ of an (N, K, D) RS code over GF(2^m), generated by $g^{\mathrm{RS}}(x)$. For each fixed $j, 0 \leq j \leq m-1$, the word $(c_{0,j}, c_{1,j}, c_{2,j}, \ldots, c_{N-1,j})$ belongs to a length-N binary cyclic code, known as the j^{th} *coordinate code.* Let $g^{\mathrm{cc}}(x) \in$ GF(2)[x] be the highest degree polynomial dividing $g^{\mathrm{RS}}(x)$. Let $g^{\mathrm{pc}}(x) = g^{\mathrm{RS}}(x)/g^{\mathrm{cc}}(x)$. It can be shown [2] that,

Theorem 1 *Each coordinate code is a subcode of the binary BCH code generated by* $g^{\mathrm{cc}}(x)$.

By Theorem 1, each coordinate is decodable by a decoder for the binary BCH code. On receiving a word $\mathbf{r}$ over GF(2^m), the m coordinate words are decoded in parallel. The outputs are then combined to form a word over GF(2^m). Finally, the roots of $g^{\mathrm{pc}}(x)$ are used to detect residual errors. Error-trapping decodable coordinate codes are particularly interesting. For, the corresponding RS codes can be decoded by m error-trapping decoders each of which uses only binary logic. Residual errors can also be detected with a simple binary logic. This simplifies the overall decoder hardware.

Performance Under 2^m-ary symmetric independent and identically distributed errors, the decoding error rate P_E for split and BDSL decoders are shown in Figure 1. Split decoding performs poorer than BDSL decoding at low channel error rates. As the channel error rate increases, the performance of BDSL decoder deteriorates faster than that of split decoder. When the capacity is close to zero, as with deep fading or pseudonoise interference (PNI), all error patterns are roughly equi-probable. So, the larger the volume of a decoding sphere is, the poorer P_E will be. In this case, split decoder outperforms BDSL decoder for several classes of RS codes, as shown in Table 1. Thus, split decoding is suitable for low-capacity channels. Also, since split decoding can be simpler and faster than BDSL decoding, the former may be advantageous even when BDSL decoding corrects more errors.

Interleave Depth On channels where the length of burst error almost never exceeds a given limit L_b, symbol interleaving can be used to reduce P_E. Let $t_{\mathrm{RS}} = \lfloor (D-1)/2 \rfloor$ and let t_{cc} be the bit error correcting capability of the coordinate code. Let $\Delta \geq \lceil L_b / t_{\mathrm{RS}} \rceil - 1$, be the symbol interleave depth of a BDSL decoder which makes $P_E \approx 0$. The interleave depth for a split decoder without residual error detection must, therefore, be $\delta \geq \lceil (\Delta+1)t_{RS}/t_{cc} \rceil - 1$. The buffer requirement is $mN\delta$ bits. When $\Delta = 0$, the buffer for Golay coordinates is 758 bits and that for Hamming coordinates is $2^{m-2}(2^m - 1)m$ bits. With residual error detection, we may be able to get by with an even smaller buffer.

---

[1] Teleglobe Communications, Inc., North Andover, MA 01845, and Department of Electrical and Computer Engineering, Northeastern University, Boston, MA 02115. Project has been partially supported by the National Science Foundation under Grant Number NCR90031004.

[2] College of Computer Science, Northeastern University, Boston, MA 02115. Project is partially supported by the National Security Agency under Grant Number MDA904-91-H-0012. The United States Government is authorised to reproduce and distribute reprints notwithstanding any copyright notation hereon.

References

[1] Lin and Costello: *Error Control Coding: Fundamentals and Applications,* Prentice-Hall, 1983

[2] Sakakibara, et al.: "On the q-ary Expansions of Reed-Solomon Codes over GF(q^m)," *Electronics and Communications in Japan,* Part III, February 1989

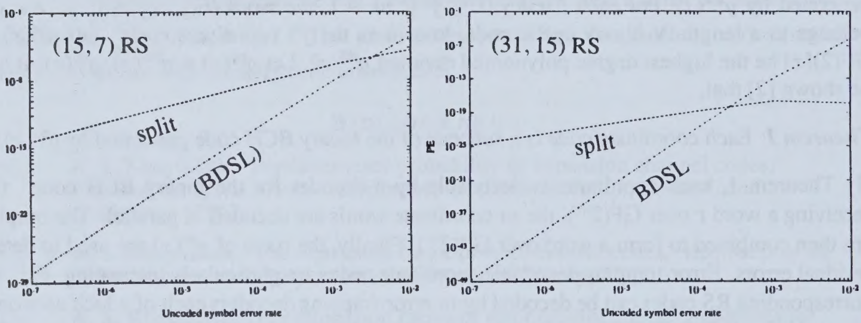


Figure 1.

Table 1: Number of Correctable Error Patterns*

Field	N	K	t_{cc}	t_{RS}	η_{split}	η_{symbol}	Which has poorer P_E in PNI?
GF(2^3)	7	3	1	2	5.1×10^2	1.1×10^3	Symbol-level decoder
GF(2^4)	15	3	3	6	1.1×10^{11}	5.9×10^{10}	Split decoder
		7	1	4	6.6×10^4	7.1×10^7	Symbol-level decoder
GF(2^5)	31	7	3	12	3.1×10^{18}	1.1×10^{26}	Symbol-level decoder
		15	1	8	3.4×10^7	7.3×10^{16}	Symbol-level decoder
GF(2^6)	21	5	2	8	1.5×10^{14}	5.1×10^{19}	Symbol-level decoder
GF(2^8)	17	5	2	6	3.2×10^{17}	3.4×10^{18}	Symbol-level decoder
GF(2^{11})	23	5	3	9	2.6×10^{36}	5.2×10^{35}	Split decoder
GF(2^{12})	7	3	1	2	6.9×10^{10}	3.5×10^8	Split decoder
	15	3	3	6	1.3×10^{33}	2.4×10^{25}	Split decoder
	15	7	1	4	2.8×10^{14}	3.8×10^{17}	Symbol-level decoder
GF(2^{32})	17	5	2	6	1.0×10^{70}	7.7×10^{61}	Split decoder

*η_{split} = size of decoding sphere for split decoding
η_{symbol} = size of decoding sphere for BDSL decoding

Sequential Decoding on Finite State Channels without Intersymbol Interference under the Pe-Criterion

Ivonete Markman[*]
ECE Department
Rose-Hulman Institute of Technology
Terre Haute, IN 47803-3999
email: markman@ee.rose-hulman.edu

John B. Anderson
ECSE Department
Rensselaer Polytechnic Institute
Troy, NY 12180-3590
email: anderson@ecse.rpi.edu

The Pe-criterion is a recent analysis [1–3] of limited search sequential decoding of tree codes based on the design condition that decoders may fail with a probability of error $Pe > 0$. The result is a definite boundary for the tree search region and a well defined estimate of the number of paths searched.

This work applied the Pe-criterion to the sequential decoding of binary tree codes on a class of finite state channels known as channels without intersymbol interference, and analyzed the implications of both hard and unquantized soft decision decoding. For these channels the state diagram constitutes a first-order Markov chain. For a channel with M possible states, and given a previous state, $S_{n-1} \in \{0, 1, \ldots, M-1\}$, the current input and output symbols are related by a set of channel transition probabilities (densities), $P(Y_n/X_n, S_{n-1})$, for the hard (unquantized soft) decision case.

It was assumed that the previous state information is available at each interval n, and the metric is given by the expression

$$\mu = \sum_{n=1}^{N} \left\{ \log_2 \left[\frac{P(Y_n/X_n, S_{n-1})}{P(Y_n/S_{n-1})} \right] - B \right\} \tag{1}$$

where N is the sequence length, B is a generic bias term and $P(.)$ represents probability (density) for the hard (unquantized soft) decision case. When $B = R$, this metric corresponds to the maximum a posteriori (MAP) decision rule given the state information.

We initially derived the shape of the search region, in a generalized distance versus depth diagram, the generalized distance being a sub-function of the metric. In such a diagram, the search region is bounded by the drop line, the set of points outside of which the correct path in the tree wanders with probability Pe or less. Its derivation comes from a special case of the Chernoff bound [3]. Calculating the drop line points required finding the maximum eigenvalue of a modified state transition matrix, which is a function of the channel parameters. Comparisons between the drop lines for channels with hard and unquantized soft decision decoding, under the same value of Pe, have shown the decrease in the search region resulting from the use of soft decision.

An estimation of the expected number of paths that a non-backtracking algorithm views within its search region was derived from an analytical method based on difference (hard decision) or integral (unquantized soft decision) equations [1,2]. This estimate was proven to be minimum when the metric bias B equals the code rate R, being proportional

---

[*] Supported by CNPq - Conselho Nacional de Desenvolvimento Científico e Tecnológico, Brazil.

		Hard Decision		Soft Decision	
R	Pe	I	II	I	II
1/2	10^{-6}	25.5	603.9	2.3	4.7
	10^{-8}	94.5	$6.4E+3$	3.8	9.9
2/3	10^{-6}	973.1	$3.8E+6$	12.3	52.7
	10^{-8}	$1.1E+4$	$6.6E+8$	31.3	217.4

Table 1: Number of paths searched for two-state channels.

Channel	q_0	q_1	p_0	p_1
I	$1.0E-3$	0.1	$1.0E-3$	0.03
I I	$3.8E-4$	$7.8E-4$	$1.0E-4$	0.02

Table 2: Channel parameters for two-state channels.

to $Pe^{(h+1)/h}$, where Pe is the desired probability of error. Solution for the value of h required finding the maximum eigenvalue of a modified state transition matrix, which is a function of R and the channel parameters.

A numerical analysis was then performed for some simple channels with only two states. These channels are equivalently described by a Gilbert-Elliott model, which may be used to represent slow-fading channels. Table 1 shows the number of paths searched for two channels, for hard and unquantized soft decision decoding, and different values of R and Pe. They are described by the transition probabilities in Table 2, where q_0 is the state transition probability associated with leaving state 0, and p_0 is the BSC channel crossover probability of state 0. Equivalent values of bit energy-to-noise ratio were calculated for p_0 and p_1, for the soft decision case. Table 1 shows that substantial savings may be achieved for these channels, when soft decision decoding is applied, especially as Pe decreases and R increases.

Overall, the numerical results for a number of channels allowed us to analyze the effect of the channel parameters in the decoding process, this time from the point of view of the number of paths searched. We observed that the channel transition probabilities (densities) have a greater effect on the channel performance than the state transition probabilities. We also observed that, as previously shown for other channels [2,3], large savings may be achieved by the use of soft decision. Furthermore, in many cases, soft decision is needed in order to assure feasibility of the limited search decoder.

LITERATURE CITED

[1] J.B. Anderson, "Sequential Decoding Based on an Error Criterion", *IEEE Transaction on Information Theory*, vol.IT-38, no.3, pp.987–1001, May 1992.

[2] I. Markman and J.B. Anderson, "Sequential Decoding on Memoryless Soft Decision Channels under the Pe-Criterion", *Proc. 1993 IEEE International Symposium on Information Theory*, January 1993, San Antonio, TX.

[3] I. Markman and J.B. Anderson, "Breadth-First Sequential Decoding on Intersymbol Interference Channels under the Pe-Criterion", *Proc. 1993 Canadian Workshop on Information Theory*, May 1993, Rockland, Ontario, Canada.

DES Supervisor Design for Hybrid Control Systems

James A. Stiver and Panos J. Antsaklis
University of Notre Dame, Notre Dame, IN 46556

Abstract

A hybrid control system consists of a continuous-time plant with a continuous state space, which is being supervised by a discrete event system. The hybrid control system also contains an interface which provides communication between the plant and controller. The interface translates the continuous-time output of the plant to an asynchronous sequence of symbols understood by the supervisor. It also translates the symbolic commands of the supervisor into a continuous input for the plant.

We present a DES model which is sufficiently flexible to represent the combined plant and interface, and use this DES model of the plant as a basis for designing the DES supervisor. This DES model is a generalization of the basic DES found in the Ramadge-Wonham framework. We extend the concepts of controllability and of the supremal controllable sublanguage and apply them to the DES model of the plant.

To design the DES supervisor for the hybrid control system, the control goals are expressed in terms of a desired language for the DES model of the plant and the controllability concepts are applied. Following analysis of the DES model of the plant, a DES supervisor is constructed which is then used to control the hybrid control system.

1 Introduction

Hybrid control systems contain both continuous and discrete dynamics. The continuous dynamics are modeled by a set of ordinary differential equations and form the portion of the system referred to as the plant. The discrete dynamics comprise a discrete event system (DES) which, in the case of this paper, is modeled as a finite automaton and referred to as the controller. The names plant and controller reflect the fact that it is common for the discrete portion of the system to be designed as a controller for the continuous portion, however, there is generally a continuous-time controller included in the plant of a hybrid system.

A home heating and cooling system is a familiar example of a hybrid control system. The plant consists of the furnace and air conditioner which, along with the heat flow characteristics form a continuous-time system. These are in turn controlled by the thermostat which is a discrete event system. Other examples of hybrid control systems can be found in chemical process controls and flexible manufacturing systems. Recently, efforts have been made to study hybrid systems in a unified, analytical way, [1–11].

In this paper, hybrid control systems are modeled using a three level structure. The top level is the finite automaton which represents the DES controller, the middle level is an interface, and the bottom level is a continuous-time plant. The interface is necessary in order for the plant and controller to interact. This is because the plant's input and output consist of continuous, real valued signals, while the controller's input and output

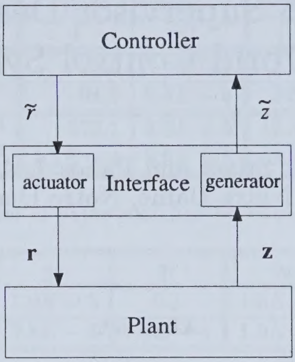


Figure 1: Hybrid Control System

are asynchronous sequences of symbols. Therefore the role of the interface is to translate between these two domains as will be described in the following section.

It is possible to combine the plant and the interface into a single entity which can be modeled as a nondeterministic finite automaton. This combination allows the entire system to be modeled as a pair of interacting discrete event systems and thus paves the way for the use of existing DES theoretic techniques for analysis and controller design.

This paper describes the hybrid control system model and associated DES models and then presents extensions to some of the DES tools developed by Ramadge and Wonham which allow these tools to be applied to hybrid control systems. In particular, a new result concerning the important concept of controllability is introduced, thus opening the way for the application of controller design methods. Two examples are included to illustrate these ideas. The design of hybrid control system controllers is also shown.

2 Hybrid Control System Model

A hybrid control system, can be divided into three parts as shown in Figure 1. The models we use for each of these three parts, as well as the way they interact are now described.

2.1 Plant

The system to be controlled, called the plant, is modeled as a time-invariant, continuous-time system. This part of the hybrid control system contains the entire continuous-time portion of the system, possibly including a continuous-time controller. Mathematically, the plant is represented by the familiar equations

$$\dot{\mathbf{x}} = f(\mathbf{x},\mathbf{r}) \quad (1)$$

$$\mathbf{z} = g(\mathbf{x}) \quad (2)$$

where $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{r} \in \mathbb{R}^m$, and $\mathbf{z} \in \mathbb{R}^p$ are the state, input, and output vectors respectively. $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ and $g : \mathbb{R}^n \to \mathbb{R}^p$ are functions. For the purposes of this work we assume that $\mathbf{z} = \mathbf{x}$. Note that the plant input and output are continuous-time vector valued signals. Bold face letters are used to denote vectors and vector valued signals.

2.2 Controller

The controller is a discrete event system which is modeled as a deterministic automaton. This automaton can be specified by a quintuple, $\{\tilde{S}, \tilde{Z}, \tilde{R}, \delta, \phi\}$, where $\tilde{S}$ is the (possibly infinite) set of states, $\tilde{Z}$ is the set of *plant symbols*, $\tilde{R}$ is the set of *controller symbols*, $\delta : \tilde{S} \times \tilde{Z} \to \tilde{S}$ is the state transition function, and $\phi : \tilde{S} \to \tilde{R}$ is the output function. The symbols in set $\tilde{R}$ are called controller symbols because they are generated by the controller. Likewise, the symbols in set $\tilde{Z}$ are called plant symbols and are generated by the occurrence of events in the plant. The action of the controller can be described by the equations

$$\begin{aligned} \tilde{s}[n] &= \delta(\tilde{s}[n-1], \tilde{z}[n]) && (3) \\ \tilde{r}[n] &= \phi(\tilde{s}[n]) && (4) \end{aligned}$$

where $\tilde{s}[n] \in S, \tilde{z}[n] \in \tilde{Z}$, and $\tilde{r}[n] \in \tilde{R}$. The index n specifies the order of the symbols in a sequence. The input and output signals associated with the controller are asynchronous sequences of symbols, rather than continuous-time signals. Notice that there is no delay in the controller. The state transition, from $\tilde{s}[n-1]$ to $\tilde{s}[n]$, and the controller symbol, $\tilde{r}[n]$, occur immediately when the plant symbol $\tilde{z}[n]$ occurs.

Tildes are used to indicate that the particular set or signal is made up of symbols. For example, $\tilde{Z}$ is the set of plant symbols and $\tilde{z}$ is a sequence of plant symbols. An argument in brackets, e.g. $\tilde{z}[n]$, represents the nth symbol in the sequence $\tilde{z}$. A subscript, e.g. $\tilde{z}_i$, is used to denote a particular symbol from a set.

2.3 Interface

The controller and plant cannot communicate directly in a hybrid control system because each utilizes a different type of signal. Thus an interface is required which can convert continuous-time signals to sequences of symbols and vice versa. The interface consists of two rather simple maps, α and γ.

The first map, called the *generator*, $\alpha : \mathbb{R}^n \times \mathbb{R}^n \to \tilde{Z}$, generates a sequence of plant symbols based upon the state of the plant. The generator, α, is based on a set of functions, $h_i(\mathbf{x})$, each of which forms a boundary in the state space. Together, these boundaries partition the state space into a number of regions. Whenever the state of the plant crosses a boundary, a *plant event* has occurred. Associated with each plant event is a plant symbol which is generated by the generator when the plant event is detected. Thus, the generator detects boundary crossings and reports each crossing via a plant symbol.

For a mathematical treatment of the generator we start by defining the sequence $\tau[n]$ which gives the times at which the plant events occur.

$$\begin{aligned} \tau[0] &= 0 && (5) \\ \tau[n] &= \inf\{t > \tau[n-1] : \\ &\qquad \exists i, h_i(\mathbf{x}(t)) \cdot h_i(\mathbf{x}(\tau[n-1] + \epsilon)) < 0\} && (6) \end{aligned}$$

With the plant event times defined, the sequence of plant symbols can be defined as follows.

$$\tilde{z}[n] = \begin{cases} \tilde{z}_i^+ & \text{if} \quad h_i(\mathbf{x}(\tau[n] + \epsilon)) > 0 \\ \tilde{z}_i^- & \text{if} \quad h_i(\mathbf{x}(\tau[n] + \epsilon)) < 0 \end{cases} \qquad (7)$$

where i is from equation 6 and ϵ is an infinitesimal positive real number. So we see that the set of plant symbols consists of two symbols for each boundary, which indicate which boundary has been crossed and in which direction.

The second map, called the *actuator*, $\gamma : \tilde{R} \to \mathbb{R}^m$, converts a sequence of controller symbols to a piecewise constant plant input as follows

$$\mathbf{r}(t) = \gamma(\tilde{r}[n]) \tag{8}$$

where

$$n = \max\{m : \tau[m] < t\} \tag{9}$$

and $\tau[m]$ is the sequence defined in equation 6. The plant input, $\mathbf{r}(\mathbf{t})$, can only take on certain specific values, where each value is associated with a particular controller symbol. Thus the plant input is a piecewise constant signal which changes only when controller symbols occur.

Part of this event characterization was suggested by M. Lemmon. In earlier formulations [10, 11] events were associated with regions rather than boundaries.

2.4 DES Plant Model

If the plant and interface of a hybrid control system are viewed as a single component, this component behaves like a discrete event system. It is advantageous to view a hybrid control system this way because it allows it to be modeled as two interacting discrete event systems which are more easily analyzed than the system in its original form. The discrete event system which models the plant and interface is called the *DES Plant Model* and is modeled as an automaton similar to the controller. The automaton is specified by a quintuple, $\{\tilde{P}, \tilde{Z}, \tilde{R}, \psi, \xi\}$, where $\tilde{P}$ is the set of states, $\tilde{Z}$ and $\tilde{R}$ are the sets of plant symbols and controller symbols, $\psi : \tilde{P} \times \tilde{Z} \to \tilde{P}$ is the state transition function, and $\xi : \tilde{P} \times \tilde{R} \to \mathbf{P}(\tilde{Z})$ is the enabling function.

The behavior of the DES plant model is as follows

$$\begin{aligned} \tilde{z}[n+1] &\in \xi(\tilde{p}[n], \tilde{r}[n]) \qquad (10) \\ \tilde{p}[n] &= \psi(\tilde{p}[n-1], \tilde{z}[n]) \qquad (11) \end{aligned}$$

where $\tilde{p}[n] \in \tilde{P}, \tilde{r}[n] \in \tilde{R}$, and $\tilde{z}[n] \in \tilde{Z}$. The enabling function defines which events are enabled in a given state and input. The state transition function defines the state which results following the occurrence of an event. The state transition function, ψ, is a partial function because some events are never enabled from a given state. In general, more than one event will be enabled for a given state and input, and therefore the DES plant may be nondeterministic.

If we let $\tilde{Z}^*$ be the set of all strings of plant symbols, then the state transition function can also be used to generate the plant state after a string of events.

$$\tilde{p}[n] = \psi(\tilde{p}[0], \tilde{w}) \tag{12}$$

where $\tilde{w} \in \tilde{Z}^*$ of length n.

The set of states, $\tilde{P}$, of the DES plant model is based upon the partition realized in the generator. Specifically, each state in $\tilde{P}$ corresponds to a region, in the state space of the continuous-time plant.

As mentioned above, the DES plant model will generally be nondeterministic which can be a problem from the standpoint of control. To avoid this problem the boundaries which define the plant events must be chosen in a way which yields a manageable DES plant model. This is not the topic of this paper, however a treatment of this issue can be found in [12] under quasideterminism.

3 DES Models for Hybrid Control Systems

In the previous section, a DES plant model was described which allows the plant and interface of a hybrid control system to be modeled as a DES. Together with the controller, the DES plant forms a DES model for the hybrid control system called the HDES. Before existing DES techniques developed in the Ramadge-Wonham framework can be extended, certain differences must be dealt with. The models used in the Ramadge-Wonham framework differ from the HDES and therefore the theorems and techniques cannot be applied directly, but must be adapted. In this section the differences will be outlined and explained.

The Ramadge-Wonham model (RWM) consists of two interacting DES's called the generator and supervisor. The generator is analogous to the DES plant model of a HDES and the supervisor is analogous to the controller. Using our notation, the generator of the RWM is an automaton consisting of a state set, $\tilde{P}$, a set of event labels or symbols, $\tilde{Z}$, and a state transition function, $\psi : \tilde{P} \times \tilde{Z} \to \tilde{P}$. So far this is the same as our DES plant model, the difference appears in the way events are enabled.

The events in the RWM are divided into two sets, those which are controllable and those which are uncontrollable: $\tilde{Z} = \tilde{Z}_c \cup \tilde{Z}_u$. The events in $\tilde{Z}_c$ can be individually enabled or disabled by commands from the supervisor, while the events in $\tilde{Z}_u$ are always enabled. The supervisor can enable any subset of $\tilde{Z}_c$, and it's ability to enable and disable events is not changed by the state of the generator.

This is in contrast to our DES plant model where each command (controller symbol) from the DES controller enables a particular subset of $\tilde{Z}$ determined by ξ, and this subset depends not only on the controller symbol but also the present state of the DES plant model. In addition, there is no guarantee that any arbitrary subset of $\tilde{Z}$ can be enabled while other plant events are not enabled. The general inability to enable events individually and the enabling function's dependence upon the state of the DES plant model, are what differentiate the HDES from the RWM DES of the Ramadge-Wonham framework.

The reason for the differences between the RWM generator and the DES plant model is that the DES plant model is not representing a true discrete event system but rather a continuous-state plant with an interface. This means that each state in the DES plant model corresponds to a subset of states in the actual plant. Since continuous-state systems will react differently to inputs depending on their current state, the DES plant model will react differently to inputs depending on its current state. A consequence of the actual plant having a continuous rather than a discrete state space is that an event consists of a subset of the state space rather than one specific state transition. This generally prevents the individual enabling of particular events.

3.1 Controllability of the HDES

The behavior of a DES can be characterized by the set of possible event sequences which it can generate. This set is referred to as the language of the DES, denoted L, and defined as

$$L = \{w : w \in \tilde{Z}^*, \phi(p_0, w) \text{ is defined}\} \tag{13}$$

where $\tilde{p}_0 \in \tilde{P}$. This section outlines new results about the controllability of languages in a HDES.

When a DES is controlled by another system such as a RWM supervisor or a DES controller, the DES will generate event strings which lie in a subset of its language. If we denote the language of the DES under control as L_c then $L_c \subset L$. A theorem has been developed to determine whether a given RWM generator can be controlled to a desired language [13]. That is, whether it is possible to design a controller such that the DES will be restricted to some target language K. The theorem states that it is possible if

$$\bar{K}\tilde{Z}_u \cap L \subset \bar{K} \tag{14}$$

where $\bar{K}$ represents the set of all prefixes of K. When this condition is met for a generator of language L, the language K is said to be controllable, and a controller can be designed which will restrict the generator to the language K. This condition requires that if an uncontrollable event occurs after the generator has produced a prefix of K, the resulting string must still be a prefix of K because the uncontrollable event cannot be prevented.

For the HDES, the language K is controllable if

$$\forall w \in \bar{K} \; \exists \, \tilde{r} \in \tilde{R} \ni w\xi(\psi(\tilde{p}_0, w), \tilde{r}) \subset \bar{K} \tag{15}$$

This condition requires that for every prefix of the desired language, K, there exists a control, $\tilde{r}$, which will enable only events which will cause string to remain in K.

Theorem 1 *If K is prefix closed and controllable according to 15, then a DES controller can be designed which will control the DES plant model to the language K.*

Proof: The proof can be found in [14].

Since the DES plant model can be seen as a generalization of the RWM, the conditions in 15 should reduce to those of 14 under the appropriate restrictions. This is indeed the case.

If the desired behavior (i.e. language) is not attainable for a given controlled DES, it may be possible to find a more restricted behavior which is. If so, the least restricted behavior is desirable. [13] and [15] describe and provide a method for finding this behavior which is referred to as the *supremal controllable sublanguage*, $K^\uparrow$, of the desired language. The supremal controllable sublanguage is the largest subset of K which can be attained by a controller. $K^\uparrow$ can be found via the following iterative procedure.

$$\begin{align} K_0 &= K \tag{16}\\ K_{i+1} &= \{w : w \in \bar{K}, \bar{w}\tilde{Z}_u \cap L \subset \overline{K_i}\} \tag{17}\\ K^\uparrow &= \lim_{i \to \infty} K_i \tag{18} \end{align}$$

For hybrid control systems, the supremal controllable sublanguage of the DES plant model can be found by a similar iterative scheme.

$$\begin{align} K_0 &= K \tag{19}\\ K_{i+1} &= \{w : w \in \bar{K}, \forall v \in \overline{w} \; \exists \, \tilde{r} \in \tilde{R} \ni \notag\\ &\qquad v\xi(\psi(\tilde{p}_0, v), \tilde{r}) \in \overline{K_i})\} \tag{20}\\ K^\uparrow &= \lim_{i \to \infty} K_i \tag{21} \end{align}$$

Theorem 2 *For a DES plant model and language K, $K^\uparrow$ is controllable and contains all controllable sublanguages of K.*

Proof: The proof can be found in [14].

With this controllability result we are able to design controllers for hybrid control systems. The procedure is illustrated in the second example.

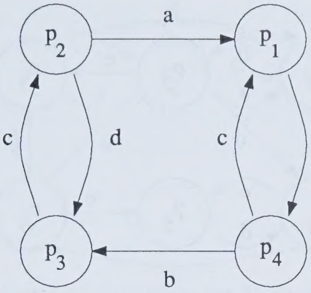


Figure 2: DES Plant Model for Example 1

4 Examples

This section contains two examples which illustrate the material of this paper.

4.1 Example 1 - Double Integrator

Using the double integrator example from [10], we have the following plant

$$\dot{\mathrm{x}} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \mathrm{x} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \mathrm{r} \tag{22}$$

In the interface, the function α partitions the state space into four regions as follows,

$$\begin{cases} \tilde{p}_1 & \text{if} \quad x_1, x_2 > 0 \\ \tilde{p}_2 & \text{if} \quad x_1 < 0, x_2 > 0 \\ \tilde{p}_3 & \text{if} \quad x_1, x_2 < 0 \\ \tilde{p}_4 & \text{if} \quad x_1 > 0, x_2 < 0 \end{cases}, \tag{23}$$

and the function γ provides three set points,

$$\gamma(\tilde{r}) = \begin{cases} -10 & \text{if} \quad \tilde{r} = \tilde{r}_1 \\ 0 & \text{if} \quad \tilde{r} = \tilde{r}_2 \\ 10 & \text{if} \quad \tilde{r} = \tilde{r}_3 \end{cases}, \tag{24}$$

Combining the plant and interface, we obtain the DES plant model. This DES is represented by the automaton in figure 2, and explicitly by the following sets and functions.

$$\tilde{P} = \{\tilde{p}_1, \tilde{p}_2, \tilde{p}_3, \tilde{p}_4\} \tag{25}$$

$$\tilde{Z} = \{\tilde{a}, \tilde{b}, \tilde{c}, \tilde{d}\} \tag{26}$$

$$\tilde{R} = \{\tilde{r}_1, \tilde{r}_2, \tilde{r}_3\} \tag{27}$$

$$\psi(\tilde{p}_1, \tilde{d}) = \tilde{p}_4 \quad \psi(\tilde{p}_3, \tilde{c}) = \tilde{p}_2 \tag{28}$$

$$\psi(\tilde{p}_2, \tilde{a}) = \tilde{p}_1 \quad \psi(\tilde{p}_4, \tilde{c}) = \tilde{p}_1 \tag{29}$$

$$\psi(\tilde{p}_2, \tilde{d}) = \tilde{p}_3 \quad \psi(\tilde{p}_4, \tilde{b}) = \tilde{p}_3 \tag{30}$$

$$\xi(\tilde{p}_1, \tilde{r}_1) = \{\tilde{d}\} \quad \xi(\tilde{p}_3, \tilde{r}_3) = \{\tilde{c}\} \tag{31}$$

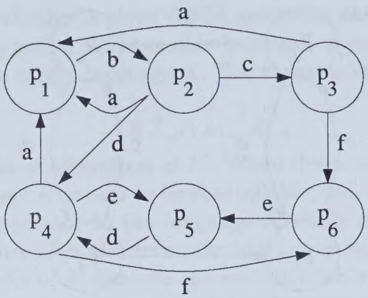


Figure 3: DES Plant Model for Example 2

$$\xi(\tilde{p}_2,\tilde{r}_1)=\{\tilde{a},\tilde{d}\} \quad \xi(\tilde{p}_4,\tilde{r}_1)=\{\tilde{b}\} \tag{32}$$
$$\xi(\tilde{p}_2,\tilde{r}_2)=\{\tilde{a}\} \quad \xi(\tilde{p}_4,\tilde{r}_2)=\{\tilde{b}\} \tag{33}$$
$$\xi(\tilde{p}_2,\tilde{r}_3)=\{\tilde{a}\} \quad \xi(\tilde{p}_4,\tilde{r}_3)=\{\tilde{b},\tilde{c}\} \tag{34}$$

The values, for which ψ has not been stated, are undefined, and the values for which ξ has not been stated are the empty set.

The language generated by this automaton is $L=\overline{((dc)^*+db(cd)^*ca)^*}$. If we want to drive the plant in clockwise circles, then the desired language is $K=\overline{(dbca)^*}$. It can be shown that this K satisfies equation (15) and therefore according to theorem 1, a controller can be designed to achieve the stated control goal.

4.2 Example 2 - A More Complex DES Plant Model

This example has a richer behavior and will illustrate the generation of a supremal controllable sublanguage. We start immediately with the DES plant model shown in figure 3. The enabling function, ξ, is given by the following table.

ξ	$\tilde{r}_1$	$\tilde{r}_2$	$\tilde{r}_3$	$\tilde{r}_4$
$\tilde{p}_1$	$\emptyset$	$\{\tilde{b}\}$	$\{\tilde{b}\}$	$\emptyset$
$\tilde{p}_2$	$\{\tilde{a}\}$	$\{\tilde{a},\tilde{d}\}$	$\{\tilde{c}\}$	$\{\tilde{a},\tilde{c},\tilde{d}\}$
$\tilde{p}_3$	$\{\tilde{a}\}$	$\{\tilde{f}\}$	$\emptyset$	$\{\tilde{a},\tilde{f}\}$
$\tilde{p}_4$	$\{\tilde{a}\}$	$\{\tilde{f}\}$	$\{\tilde{a},\tilde{f}\}$	$\{\tilde{a},\tilde{e},\tilde{d}\}$
$\tilde{p}_5$	$\emptyset$	$\{\tilde{d}\}$	$\{\tilde{d}\}$	$\{\tilde{d}\}$
$\tilde{p}_6$	$\{\tilde{e}\}$	$\{\tilde{e}\}$	$\{\tilde{e}\}$	$\{\tilde{e}\}$

(35)

The language generated by this DES is $L=\overline{L_m}$ where

$$L_m=(b(a+d\sigma^*a+c(a+fed\sigma^*a)))^* \tag{36}$$

and $\sigma=((e+fe)d)$. Suppose we want to control the DES so that it never enters state $\tilde{p}_5$ and can always return to state $\tilde{p}_1$. The desired language is therefore

$$K=\overline{(a+b+c+d+f)^*a} \tag{37}$$

In this example, the language K is not controllable. This can be seen by considering the string $bcf \in K$, for which there exists no $\tilde{r} \in \tilde{R}$ which will prevent the DES from deviating from K by generating e and entering state $\tilde{p}_5$.

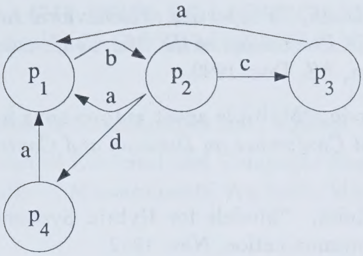


Figure 4: Controller for Example 2

Since K is not controllable, we find the supremal controllable sublanguage of K as defined in equation 21. The supremal controllable sublanguage is

$$K^{\uparrow} = K_1 = \overline{(a+b+c+d+f)^* a - (bcfed\sigma^* a)^*} \tag{38}$$

Obtaining a DES controller once the supremal controllable sublanguage has been found is straight forward. The controller is a DES whose language is given by $K^{\uparrow}$ and the output of the controller in each state, $\phi(\tilde{s})$, is the controller symbol which enables only transitions which are found in the controller. The existence of such a controller symbol is guaranteed by the fact that $K^{\uparrow}$ is controllable. For Example 2, the controller is shown in Figure 4 and it's output function, ϕ, is as follows:

$$\phi(\tilde{s}_1) = \tilde{r}_2 \qquad \phi(\tilde{s}_2) = \tilde{r}_4 \tag{39}$$

$$\phi(\tilde{s}_3) = \tilde{r}_1 \qquad \phi(\tilde{s}_4) = \tilde{r}_1 \tag{40}$$

5 Conclusion

We have extended the concepts of the controllability and the supremal controllable sublanguage of a given language, so that they may be applied to hybrid control systems. Various controller design techniques, developed for DES's, can now be applied to HCS's.

The relationship between the controllability of the continuous-time plant and the controllability of the DES plant model has not been fully examined. This issue needs further study. Nevertheless, it is not difficult to see that the two controllability concepts are related. If the continuous-time plant is not controllable, then it is likely that certain event sequences will not be achievable and this certainly affects the controllability of the DES plant model.

It should be noted that the problem of how properties of the continuous-time plant, such as controllability, observability, and stability, affect similar properties in DES plant model is currently under further investigation by the authors.

Acknowledgement - the partial support of NSF grant MSS-9216559 is acknowledged.

References

[1] A. Benveniste and P. Le Guernic, "Hybrid dynamical systems and the signal language", *IEEE Transactions on Automatic Control*, vol. 35, no. 5, pp. 535–546, May 1990.

[2] A. Gollu and P. Varaiya, "Hybrid dynamical systems", In *Proceedings of the 28th Conference on Decision and Control*, pp. 2708–2712, Tampa, FL, Dec. 1989.

[3] L. Holloway and B. Krogh, "Properties of behavioral models for a class of hybrid dynamical systems", In *Proceedings of the 31st Conference on Decision and Control*, pp. 3752–3757, Tucson, AZ, Dec. 1992.

[4] W. Kohn and A. Nerode, "Multiple agent autonomous hybrid control systems", In *Proceedings of the 31st Conference on Decision and Control*, pp. 2956–2966, Tucson, AZ, Dec. 1992.

[5] A. Nerode and W. Kohn, "Models for Hybrid Systems: Automata, Topologies, Stability", Private Communication, Nov. 1992.

[6] M. D. Lemmon, J. A. Stiver, and P. J. Antsaklis, "Learning to coordinate control policies of hybrid systems", In *Proceedings of the American Control Conference*, pp. 31–35, San Francisco, CA, June 1993.

[7] K. M. Passino and U. Ozguner, "Modeling and analysis of hybrid systems: Examples", In *Proceedings of the 1991 IEEE International Symposium on Intelligent Control*, pp. 251–256, Arlington, VA, Aug. 1991.

[8] P. Peleties and R. DeCarlo, "A modeling strategy with event structures for hybrid systems", In *Proceedings of the 28th Conference on Decision and Control*, pp. 1308–1313, Tampa, FL, Dec. 1989.

[9] J. A. Stiver and P. J. Antsaklis, "A novel discrete event system approach to modeling and analysis of hybrid control systems", In *Proceedings of the Twenty-Ninth Annual Allerton Conference on Communication, Control, and Computing*, University of Illinois at Urbana-Champaign, Oct. 1991.

[10] J. A. Stiver and P. J. Antsaklis, "Modeling and analysis of hybrid control systems", In *Proceedings of the 31st Conference on Decision and Control*, pp. 3748–3751, Tucson, AZ, Dec. 1992.

[11] J. A. Stiver and P. J. Antsaklis, "State space partitioning for hybrid control systems", In *Proceedings of the American Control Conference*, pp. 2303–2304, San Francisco, California, June 1993.

[12] P. J. Antsaklis, M. D. Lemmon, and J. A. Stiver, "Hybrid system modeling and event identification", Technical Report of the ISIS Group ISIS-93-002, University of Notre Dame, Notre Dame, IN, January 1993.

[13] P. J. Ramadge and W. M. Wonham, "Supervisory control of a class of discrete event processes", Systems Control Group Report 8515, University of Toronto, Toronto, Canada, Nov. 1985.

[14] P. J. Antsaklis, M. D. Lemmon, and J. A. Stiver, "Learning to be autonomous: Intelligent supervisory control", Technical Report of the ISIS Group ISIS-93-003, University of Notre Dame, Notre Dame, IN, April 1993.

[15] W. M. Wonham and P. J. Ramadge, "On the supremal controllable sublanguage of a given language", Systems Control Group Report 8312, University of Toronto, Toronto, Canada, Nov. 1983.

[16] P. Ramadge and W. M. Wonham, "The control of discrete event systems", *Proceedings of the IEEE*, vol. 77, no. 1, pp. 81–89, Jan. 1989.

A NEW APPROACH FOR SOME COMBINATORIALLY HARD STOCHASTIC OPTIMIZATION PROBLEMS[1]

Christos G. Cassandras and Vibhor Julka

Department of Electrical and Computer Engineering
University of Massachusetts, Amherst, MA 01003

ABSTRACT

We present a new approach for solving a class of combinatorially hard stochastic optimization problems encountered in the design and control of discrete-event systems. The main idea of the approach is to transform the original discrete optimization problem into an auxiliary continuous optimization problem through an appropriate randomization mechanism and then solve the new problem using gradient-based techniques and Smoothed Perturbation Analysis (SPA) to estimate performance gradients from a single observed sample path. Mapping back to the original discrete decision space yields the optimal solution in that space.

1 Introduction

In this paper we propose an approach for solving some combinatorially hard stochastic optimization problems over discrete decision spaces. The difficulty in the problems we consider is twofold. First, because the controlled parameters take on *discrete* values, one cannot resort to standard optimization techniques (e.g., gradient-based algorithms). Second, because of the complexity of the stochastic models involved, there are no closed-form expressions for the functional relationship between the controlled parameters and the cost criteria of interest. As a result, one must estimate cost through simulation or direct observation of sample paths.

The area of stochastic optimization over *continuous* decision spaces is quite rich and usually involves gradient-based techniques as in several well-known stochastic approximation algorithms [6, 7]. The literature in the area of discrete stochastic optimization is relatively limited, however. Most known approaches are based on some form of random search with the added difficulty of having to *estimate* the cost function at every step when closed-form expressions are unavailable, as is often the case in the design and control of complex stochastic Discrete Event Systems (DES). Such algorithms have been proposed by Yan and Mukai [9] and Gong et al. [5].

The work in this paper was originally motivated by discrete optimization problems one encounters in scheduling packet transmissions in radio networks. Such problems can be posed in the framework of resource allocation, as described in [3]. In this paper, we shall limit ourselves to a simpler classical stochastic resource allocation problem, where there are K identical *resources* to be allocated over N user classes. Further details are provided in section 2.

Our approach is quite different than that of [9] or [5]. First, by introducing auxiliary control parameters we transform the original *discrete* optimization problem into a *con-*

---

[1] This work was supported by the Naval Research Laboratory under contract N000014-92-J-2017 and the National Science Foundation under grant EID-9212122.

tinuous optimization problem. The latter is still a hard problem because of the absence of closed-form expressions for the cost in terms of these auxiliary variables. However, we exploit recent developments in perturbation analysis (PA) (see [1, 2, 4]) in order to develop gradient estimates based on observations of a single sample path, which can then be used to iteratively solve this problem. Note that at every iteration an inverse transformation imposes an iteration on the original discrete parameters as well. When (and if) an optimal solution to the continuous optimization problem is attained, an optimal solution to the original discrete optimization problem is attained as well.

The rest of the paper is organized as follows. In section 2, we provide a statement of the stochastic resource allocation problem we consider. In section 3 we proceed to consider an auxiliary continuous optimization problem and describe the key ideas of our approach. In section 4, we derive an unbiased estimator for the gradient of the cost criterion of interest with respect to the continuous parameter we have introduced. In section 5, we present a result on convergence to the optimal point on a sample path by sample path basis. We also illustrate our methodology by briefly discussing in section 6 two applications which can be modelled as resource allocation problems.

2 Discrete Stochastic Optimization Problem

Consider a resource allocation problem where K identical resources are to be allocated to N user classes. It is assumed that a queue is associated with the ith class, $i = 1, \cdots, N$, and users in that class form an arrival stream to that queue with arbitrary interarrival time distribution. When a user accesses a resource, he may hold it for some random period of time whose distribution is also arbitrary. The problem then is to determine the number of resources to be allocated to the ith class so that some particular cost criterion is minimized (e.g., the average waiting time of users who must wait in queue before accessing a resource).

To formally state this problem, let the resources be sequentially indexed so that the "state" or "allocation" s is represented by the K-dimensional vector $[s_1, \cdots, s_K]^T$ where $s_j \in \{1, \cdots, N\}$ is the user class index assigned to resource j. Let S be the discrete finite set of feasible resource allocations

$$S = \{[s_1, \cdots, s_K] : \ \ s_j \in \{1, \cdots, N\}\}$$

By "feasible" we mean that the allocation may have to be chosen to satisfy some basic requirements, such as 'stability' or 'fairness'. For any $\mathrm{s} \in S$, let $J_i(\mathrm{s}) = E[L_i(\mathrm{s})]$ be the class i expected cost function. Then, our objective is to select $\mathrm{s} \in S$ so as to solve the following discrete optimization problem:

$$\textbf{(P1)} \qquad \min_{\mathrm{s} \in S} \sum_{i=1}^{N} \beta_i E[L_i(\mathrm{s})]$$

where β_i is a weight associated with user class i.

This is generally a difficult problem, as it involves both optimization over a discrete set and estimation of the cost under all possible allocations $\mathrm{s} \in S$. A straightforward approach for solving (**P1**) is through some form of random search where the cost is estimated at each step. However, since estimation is only possible through simulation (or direct observation) of sample paths of the process, this approach involves a tremendous amount of simulation (or direct data collection) time. Moreover, it is difficult to guarantee convergence of such an optimization scheme. The latter problem is overcome by the optimization algorithms proposed in [5, 9]; in both approaches the authors choose to optimize an alternative objective function. It is then shown that the solution to the transformed problem is also a solution to the original optimization problem. It is important to note, however, that the optimization is still done over a discrete set S. The approach we propose in this paper is quite different, as it tries to transform **(P1)** into a continuous optimization problem so as to take advantage of gradient-based techniques. Intuitively, we would like to preserve the notion of "continuing downhill in a favorable direction" so as to not only continuously provide improvement, but also accelerate the descent towards the optimum using the information contained in the magnitude of the gradient.

3 The Auxiliary Continuous Optimization Problem

Let $\theta_i = P[s_j = i]$, independent of the resource index $j = 1, ..., K$. In other words, we assume that any one of the identical K resources is allocated to user class i with probability θ_i. In this setting, the probability vector $\theta = [\theta_1, ..., \theta_N]$, with $\sum_{i=1}^{N} \theta_i = 1$, represents new control parameters which can be optimally selected so as to minimize the desired cost function. In particular, we replace problem **(P1)** by the following constrained continuous parameter optimization problem:

$$\textbf{(P2)} \qquad \min_{\theta} \sum_{i=1}^{N} \beta_i E[L_i(\theta)] \qquad s.t. \quad \sum_{i=1}^{N} \theta_i = 1$$

The relationship between the resource assignment probability vector θ and the actual resource allocation $\mathbf{s}$ can be made precise by specifying the randomization mechanism used. Thus, let $\mathbf{r} = [r_1, \cdots, r_K]$ be a K-dimensional vector of independent random variables. Although there may be flexibility in choosing the distribution of r_j, for simplicity we assume that r_j is uniformly distributed in $[0, 1]$. Then, suppose that $\mathbf{r}$ is fixed by obtaining K independent random numbers. An allocation $\mathbf{s}(\theta, \mathbf{r}) = [s_1, \cdots, s_K]$ is obtained using the standard inverse transform technique for generating random variates $s_1, ..., s_K$ as follows:

$$s_j = min\{i : r_j \leq \sum_{k=1}^{i} \theta_k, \;\; i = 1, \cdots, N\} \tag{1}$$

It is then clear that, for a given $\mathbf{r}$, by adjusting the parameters θ_i, $i = 1, ..., N$, we can affect s_j, $j = 1, ..., K$, and, consequently, control the allocation of the resources. From a notational standpoint, it will be useful to distinguish between an allocation generated through (1), which is denoted by $\mathbf{s}(\theta, \mathbf{r})$, and an allocation denoted by $\mathbf{s}$, which is simply an element of the set S.

The next step in our approach is to seek a solution of **(P2)**. This is still a hard problem, since no closed-form expression is generally available for $E[L_i(\theta)]$. Here, however, we can at least resort to sample-path-based gradient estimation techniques (e.g., perturbation analysis.) Still, one can see that because of the discontinuities in the mapping (1), the sample functions $L_i(\theta)$ will generally be discontinuous in θ as well. We will show, however, that it is possible to derive unbiased estimators of the derivatives $\partial E[L_i(\theta)]/\partial\theta_i$ using an approach similar to that of [2].

Since our goal then is to estimate $\partial E[L_i(\theta)]/\partial\theta_i$ from a single sample path, we need to generate such a sample path of the DES of interest (i.e., the process of arriving users who queue up and then hold a resource for some period of time). To do so, we need to specify an allocation $\mathbf{s}(\theta, \mathbf{r})$. Assuming that a control vector θ is specified, then a sample path is characterized by the vector $\mathbf{r}$ and we write in **(P2)** $E[L_i(\theta)] = E[L_i(\mathbf{s}(\theta, \mathbf{r}))]$. Let us now assume that the derivatives $\partial E[L_i(\mathbf{s}(\theta, \mathbf{r}))]/\partial \theta_i$ can be estimated from a sample path generated under a specific $\mathbf{r}$. For the time being, let us ignore the issue of how to estimate them and use D_i to denote the ith derivative estimate. We are then in a position to summarize our approach by means of the following optimization scheme **(S1)**:

1. Initially select some $\mathbf{r}$ and $\theta^{(0)}$. Hence, determine an initial allocation $\mathbf{s}^{(0)} = \mathbf{s}(\theta^{(0)}, \mathbf{r})$ through (1).

2. Observe a sample path under $\mathbf{s}(\theta^{(0)}, \mathbf{r})$ and estimate the derivatives $D_i^{(0)}$ for all $i = 1, \cdots, N$.

3. For any $m = 0, 1, \cdots$, iterate as follows:

$$\theta_i^{(m+1)} = \theta_i^{(m)} + \eta_m D_i^{(m)} \tag{2}$$

$$s_j^{(m+1)} = min\{i : r_j \leq \sum_{k=1}^{i} \theta_k^{(m+1)}, \ \ i = 1, \cdots, N\} \tag{3}$$

   with some provisions to ensure that $\sum_{i=1}^{N} \theta_i^{(m)} = 1$ for all $m = 0, 1, \cdots$.

4. Observe a sample path under $\mathbf{s}(\theta^{(m)}, \mathbf{r})$ to estimate the derivatives $D_i^{(m)}$ for all $i = 1, \cdots, N$, and repeat step 3 until the sequence $\theta^{(m)}$, $m = 0, 1, \cdots$ converges to some $\theta^*(\mathbf{r})$, and hence an allocation $\mathbf{s}^* = \mathbf{s}(\theta^*, \mathbf{r})$.

Of course, (2) is a standard stochastic approximation scheme driven by the derivative estimates D_i^m with $\{\eta_m\}$ an appropriately selected step size sequence. The important observation here, however, is that after an update in the auxiliary vector $\theta^{(m)}$, *the allocation is also updated* using the transformation defined in (1). Moreover, note that $\mathbf{r}$ remains fixed throughout the process.

There are now several crucial issues that need to be investigated. First, we must see how to obtain derivative estimates $D_i^{(m)}$ from a single sample path. Note that even if $L_i(\mathbf{s})$ is a deterministic cost for user class i under allocation $\mathbf{s}$, our randomization mechanism turns $D_i^{(m)}$ into a stochastic derivative. Second, in dealing with iterative schemes of the type (2), one is usually concerned with the conditions under which $\{\theta^{(m)}\}$, $m = 0, 1, \cdots$ indeed converges to the optimal probability vector. In our approach, however, it is the iterative scheme (3) that is of interest; therefore, the main issue is to identify a "stopping condition" in (2) such that the resulting $\theta^*(\mathbf{r})$ yields an optimal allocation $\mathbf{s}(\theta^*, \mathbf{r})$.

4 Gradient Estimation

Let us begin by addressing the issue of gradient estimation. In particular, we are interested in deriving estimators for $\partial E[L_i(\mathbf{s}(\theta, \mathbf{r}))]/\partial \theta_i$ based on a sample path obtained when $\mathbf{r}$ is fixed. We begin by defining

$$n_i = \sum_{j=1}^{K} \mathbf{1}[s_j = i] \qquad i = 1, \cdots, N \tag{4}$$

where $\mathbf{1}(\cdot)$ is the standard indicator function. Thus, n_i is simply the number of resources allocated to user class i under some allocation $\mathbf{s}$. We shall now make the following simplifying assumption for our problem:

- *A1:* $L_i(\mathrm{s})$ depends only on the number of resources assigned to class i, i.e., $L_i(\mathrm{s}) = L_i(n_i)$.

Under this assumption, we can see that an allocation written as the K-dimensional vector $\mathrm{s} = [s_1, \cdots, s_K]$, can be replaced by the N-dimensional vector $\mathrm{s} = [n_1, \cdots, n_N]$. In addition, we can consider any one user class in isolation, i.e., dropping the subscript i, let n be the number of resources assigned to this class and $L(n, \omega)$ the corresponding cost in a sample path denoted by ω, with the underlying probability space defined as in [3]. To determine n we use a randomization mechanism parameterized by a scalar θ. The jth resource is allocated to this class as long as $\rho_j \leq \theta$ where ρ_j is a random variable uniformly distributed in $[0, 1]$. Then, define the Boolean variable $u_j(\theta)$

$$u_j(\theta) = \begin{cases} 1 & \text{if } \rho_j \leq \theta \\ 0 & \text{otherwise} \end{cases}$$

It follows that a characterization of the sample path is the set $A(\theta) = \{j : u_j(\theta) = 1\}$, and that $n = |A(\theta)|$, the cardinality of this set.

To determine the derivative of $E[L(\theta, \omega)]$ with respect to θ, consider a perturbation of magnitude $|\Delta\theta|$. Depending on the sign of $\Delta\theta$, we can define the right and left derivatives respectively. Let us limit ourselves to the case $\Delta\theta > 0$ and the left derivative

$$\begin{aligned} \frac{\partial}{\partial\theta} E[L(\theta, \omega)] &= \lim_{\Delta\theta \to 0} \frac{E[L(\theta, \omega)] - E[L(\theta - \Delta\theta, \omega)]}{\Delta\theta} = \lim_{\Delta\theta \to 0} \frac{E[\Delta L(\theta, \omega)]}{\Delta\theta} \\ &= \lim_{\Delta\theta \to 0} E\left[\frac{E[\Delta L(\theta, \omega)|A(\theta)]}{\Delta\theta}\right] \end{aligned} \tag{5}$$

The conditioning step above is typical in Smoothed Perturbation Analysis (SPA) where our goal is to replace a discontinuous sample function like $L(\theta, \omega)$ with the smoother function $E[L[(\theta, \omega)|A(\theta)]$. Now, in a sample path resulting from a perturbation $-\Delta\theta$ it is clear that the jth resource remains allocated to the user class as long as $\rho_j < \theta - \Delta\theta$. Also, note that if the jth resource is not originally allocated to this user class, then the same is true in the perturbed path, since $\rho_j > \theta > \theta - \Delta\theta$. Omitting ω for simplicity, let

$$\Delta u_j = u_j(\theta - \Delta\theta) - u_j(\theta); \quad \Delta u_j \in \{-1, 0\}$$

where '$\Delta u_j = -1$' implies that the jth resource is no longer allocated to the user class in the perturbed sample path. Moreover, let

$$\Delta u^k = \underbrace{[\Delta u_i = -1, \cdots, \Delta u_j = -1]}_{k} \tag{6}$$

and

$$\Delta L(n) = L(n) - L(n-1), \qquad n = 1, \cdots, K \tag{7}$$

With this notation in mind, we can establish the following result.

Theorem 1

$$E\left[\frac{n}{\theta}\Delta L(n)\right] = \frac{\partial}{\partial\theta} E[L(\theta, \omega)] \tag{8}$$

provided that $|E[\Delta L(n)|\Delta u^k, A(\theta)]| \leq C(A(\theta))$, $k \geq 2$, *for some random variable* $C(A(\theta))$ *with* $E[C(A(\theta))] < \infty$.

Proof: The proof is along the same lines as a similar result in [3] and only a sketch is given here. Let A be the set (in increasing order) of all resource indices assigned to the user class in the nominal system (clearly, A is sample-path-dependent). Denote the first and last elements of the set $A(\theta)$ by f_A, l_A. Omitting the arguments θ, ω for simplicity, we have:

$$\begin{aligned} E[\Delta L|A] &= \sum_{i\in A} P[\Delta u_i = -1|A] E[\Delta L|\Delta u_i = -1, A] \; + \\ &+ \sum_{i\in A} \sum_{j\in A, j\neq i} P[\Delta u_i = -1, \Delta u_j = -1|A] E[\Delta L|\Delta u_i = -1, \Delta u_j = -1, A] \; + \cdots \\ &+ P[\Delta u_{f_A} = -1, \cdots, \Delta u_{l_A} = -1|A] E[\Delta L|\Delta u_{f_A} = -1, \cdots, \Delta u_{l_A} = -1, A] \qquad (9) \end{aligned}$$

In (9) each term on the right-hand-side represents a set of possible perturbed sample paths. For example, the first summation represents all distinct perturbed allocations each of which differs from the nominal one by one removed resource. Similarly the second term considers all perturbed allocations that differ from the nominal in two distinct resources. Since the resource allocation mechanism is purely random with probability θ, we have $P[\Delta u^k] = (\Delta\theta/\theta)^k$, $k = 1, \cdots, |A|$. Thus, in (9) all sums but the first contribute terms of order higher than $\Delta\theta$. Under the technical condition stated in the theorem, dividing by $\Delta\theta$ and taking the limit in (5) gives the following expression for the derivative:

$$\frac{\partial}{\partial\theta} E[L(\theta,\omega)] = E\left[\frac{1}{\theta}\sum_{i\in A} E[\Delta L|\Delta u_i = -1, A]\right]$$

Using *A1*, the summation can be replaced by $|A| = n$. Finally, taking an expectation of the bracketed term leads to (8). ∎

Thus, $(n/\theta)\Delta L(n)$ is an unbiased estimator of $\partial E[L(\theta,\omega)]/\partial\theta$ under some mild conditions on the cost function. Note that the set $A(\theta)$ is directly observed whenever one observes a sample path used to obtain $L(n)$. The question then is whether it is possible to also obtain $L(n-1)$ from the same sample path, so as to obtain $\Delta L(n)$. It has been shown that this is indeed possible for several cost functions of interest, such as mean waiting time of users or blocking probability when a user does not gain access to a resource by a certain amount of time (and is consequently "blocked"). This is done using the so-called marked/phantom estimators derived in [2, 3]. We will therefore assume that this is the case and proceed to the next issue of interest in our optimization scheme.

5 Convergence to a Global Optimum

Let us return to the iterative scheme (2) in section 3. Based on Theorem 1 and the assertion that $\Delta L(n)$ can be obtained along a single observed sample path, we can use

$$D_i^{(m)} = \frac{n_i^{(m)}}{\theta_i^{(m)}} \Delta L_i(n_i^{(m)})$$

as the derivative estimate needed for each user class i. We can then proceed until the sequence $\{\theta^{(m)}\}$ converges (if it does) to some $\theta^*(\mathrm{r})$ in order to determine the resulting allocation $\mathrm{s}^* = \mathrm{s}(\theta^*, \mathrm{r})$ in (**S1**). There are, however, two problems with this approach. The first problem is that the derivative estimator above has a large variance. The second problem is the convergence of $\{\theta^{(m)}\}$: not only can it be extremely slow because of the high variance of $D_i^{(m)}$, but, as already pointed out in section 3, convergence of this sequence may not be necessary to guarantee that the optimal allocation has been reached. It is easy to see, for instance, that a point s may be reached and while the vector θ continues to vary through (2), no further change may result through (3).

Thus, we make two modifications to the optimization scheme presented in section 3. First, we replace $D_i^{(m)}$ by

$$\tilde{D}_i^{(m)} = K \cdot \Delta L_i(n_i^{(m)}) \tag{10}$$

Note that $E[n_i/\theta_i] = K$, since, by definition, $\theta_i = E[n_i/K]$. Therefore, we have replaced the random variable n_i/θ_i by its expectation.

Next, rather than waiting to stop the procedure when the sequence $\{\theta^{(m)}\}$ converges, we define an alternative stopping condition as one where

$$\max\{\tilde{D}_i^{(m)},\ i = 1, \cdots, N\}$$

is minimized over the iteration index m. Note that in practice, as in all gradient-based iterative schemes, one needs to define specific stopping conditions for terminating an algorithm in finite time or preventing small oscillations around a point of convergence. In our approach, it is typical to converge relatively fast to a small set of allocations (of which one is the optimal) and then simply remain in this set for all successive iterations. In practice, it is generally easy to identify this situation and pick out the true optimal allocation.

We will refer to this modified optimization scheme as (**S2**). In addition, we make the following assumption regarding the cost functions of interest:

- *A2:* For all $i = 1, \cdots, N$, $L_i(n_i)$ is monotonically non-increasing in n_i and $L_i(n_i + 1) - L_i(n_i) \geq L_i(n_i) - L_i(n_i - 1)$. Equivalently, using (7): $\Delta L_i(n_i + 1) \geq \Delta L_i(n_i)$.

We can now establish the following result. Let us first consider a specific sample path and apply the optimization scheme (**S2**). Let θ^s be the probability vector when the stopping condition is met in (2), and let $\mathbf{s}(\theta^s, \mathbf{r})$ be the corresponding allocation. This allocation is an optimal one in the discrete space S, i.e. $\mathbf{s}(\theta^s, \mathbf{r}) = \mathbf{s}^*$, where $\mathbf{s}^*$ is independent of $\mathbf{r}$ and of the sequence $\{\theta^{(m)}\}$, $m = 0, 1, \cdots$ which leads to the stopping point in (2).

Theorem 2 *Under assumptions A1-A2, the optimization scheme (***S2***) applied to a particular sample path of the underlying system yields an allocation which is optimal in the discrete space S.*

Proof: We will limit ourselves here to the case $N = 2$; the proof for $N > 2$ is through a similar argument. For simplicity, also let $\beta_i = 1$ in (**P1**) for all $i = 1, \cdots, N$. Then, the stopping condition of (**S2**) requires that $\max\{\tilde{D}_1^{(m)}, \tilde{D}_2^{(m)}\}$ be minimized. First, an allocation $\mathbf{s}^* = [n_1^*, \cdots, n_N^*]$ is optimal if $L(\mathbf{s}^*) \leq L(\mathbf{s})$ for all feasible allocations $\mathbf{s} \in S$. In addition, let

$$B(\mathbf{s}) = \{\mathbf{s} :\ \mathbf{s} = [n_1, \cdots, n_i + 1, \cdots, n_j - 1, \cdots, n_N] \ \text{ for some } i \neq j\}$$

denote the set of feasible neighboring points to $\mathbf{s} = [n_1, \cdots, n_N]$, i.e., vectors which differ from s by $+1$ and -1 in two distinct components (recall that $n_1 + \cdots + n_N = K$). Clearly, $L(\mathbf{s}^*) \leq L(s')$ for all $s' \in B(\mathbf{s}^*)$, which can be rewritten as

$$\Delta L_i(n_i^* + 1) \geq \Delta L_j(n_j^*) \text{ for all } i \neq j \tag{11}$$

Now, for $N = 2$, minimizing $\max\{\tilde{D}_1^{(m)}, \tilde{D}_2^{(m)}\}$ implies, using (10), that the allocation $[n_1^s, n_2^s]$ at this point is such that $\max\{\Delta L_1(n_1^s), \Delta L_2(n_2^s)\}$ is minimized over all feasible allocations. We will now show that there cannot exist an allocation $[m_1, m_2]$ which is better than the allocation corresponding to our stopping criterion. Thus, let the stopping criterion result in an allocation $[n_1^s, n_2^s]$. Suppose that this allocation is non-optimal. We will show that this leads to a contradiction. Let $[m_1, m_2]$ be the true global optimum. Without loss of generality, assume that $m_1 > n_1$ and $m_2 < n_2$ with $m_1 + m_2 = n_1^s + n_2^s = K$. Consider the following two cases:

Case 1: $\Delta L_1(m_1) \geq \Delta L_2(m_2)$. In this case, the stopping condition implies

$$\max\{\Delta L_1(n_1^s), \Delta L_2(n_2^s)\} \; < \; \max\{\Delta L_1(m_1), \Delta L_2(m_2)\} \; = \; \Delta L_1(m_1) \qquad (12)$$

Clearly, at $[m_1, m_2]$ we must satisfy the local optimality condition (11), therefore $\Delta L_2(m_2 + 1) \geq \Delta L_1(m_1)$. Using *A2* and the fact that $n_2^s \geq m_2 + 1 > m_2$ we have

$$\max\{\Delta L_1(n_1^s), \Delta L_2(n_2^s)\} \geq \Delta L_2(n_2^s) \geq \Delta L_2(m_2 + 1) \geq \Delta L_1(m_1)$$

which contradicts (12).

Case 2: $\Delta L_1(m_1) < \Delta L_2(m_2)$. In this case,

$$\max\{\Delta L_1(n_1^s), \Delta L_2(n_2^s)\} \; < \; \max\{\Delta L_1(m_1), \Delta L_2(m_2)\} \; = \; \Delta L_2(m_2) \qquad (13)$$

Since $n_2^s \geq m_2 + 1 > m_2$, using *A2* we get:

$$\max\{\Delta L_1(n_1^s), \Delta L_2(n_2^s)\} \geq \Delta L_2(n_2^s) \geq \Delta L_2(m_2 + 1) \geq \Delta L_2(m_2)$$

which contradicts (13). Therefore, we have shown that if $[m_1, m_2]$ is the optimal point, then the stopping criterion at $[n_1^s, n_2^s]$ would be violated and the proof is complete. ∎

6 Applications

In this section we consider two applications which can be modelled as resource allocation problems.

Example 1. Optimal Slot Allocation in Radio Networks: This application is motivated by the need to process voice calls in an N-node multihop Radio Network (RN) [8]. For simplicity, assume that all nodes are neighbors[2] of every other node i.e the network is fully connected. The common resource (total available bandwidth) is shared through a *Time Division Multiplexing (TDM)* policy with frame length K slots such that a slot is uniquely assigned to a node. We assume a slotted time model, where voice calls arrive to node i (at rate λ_i) with general interarrival distribution, and call durations are an arbitrarily distributed integral multiples of the frame length. Finally, a call is blocked if it cannot be assigned a slot in the *TDM frame* (denoted by $\mathbf{s}$) (for more details, see [3]). The stochastic discrete optimization problem is then to select the optimal $\mathbf{s}^*$ to minimize $\sum_{i=1}^{N} \beta_i P_i(\mathbf{s})$ where $P_i(\mathbf{s})$ is the node i blocking probability.

In the simulation results shown here we consider $N = 3$ nodes, $K = 24$ slots in the frame, and a symmetrically loaded system with Poisson arrivals ($\lambda_i = 0.2 \; \forall \; i = 1, \cdot\cdot, 3$) and fixed call durations of 5 frames. Thus, even though this system is difficult to analyze, it follows from the symmetry that the optimal frame is $[8, 8, 8]$. We considered three single-run optimization experiments (with different randomization vectors $\mathbf{r}_1$, $\mathbf{r}_2$, $\mathbf{r}_3$). In each case, the initial slot assignment probability vector is $\theta^{(0)} = [0.7, 0.15, 0.15]$. Table

---

[2] A node i is said to be a neighbour of node j if node i lies within the *broadcast zone* of node j

1 shows the initial and final frames, as well as the final slot assignment probabilities, where the latter two are indexed by x, the number of iterations on θ until we first reach the optimal allocation. As noted in section 5, after this point we oscillate between the optimal and one of its neighbors.

Exp.	$\mathbf{s}^{(0)}$	$\mathbf{r}$	$\mathbf{s}^{(x)}$	$\theta^{(x)}$	x
1	$(18,3,3)$	$\mathbf{r}_3$	$(8,8,8)$	$[0.391, 0.261, 0.348]$	29
2	$(20,3,1)$	$\mathbf{r}_1$	$(8,8,8)$	$[0.292, 0.197, 0.511]$	30
3	$(21,1,2)$	$\mathbf{r}_2$	$(8,8,8)$	$[0.263, 0.185, 0.552]$	38

Table 1: Optimization of 3-node Polling System

Example 2. A Simple Deterministic Resource Allocation Problem: To illustrate the potential computational savings of our approach, we compare the computational complexity (measured in terms of cost function evaluations) of several techniques for solving a discrete optimization problem within a purely *deterministic* static framework. Obviously, however, the real potential of our optimization scheme is found in stochastic problems, where the gradients driving the iteration (1) are estimated over a *single sample path.*

We have considered the following methodologies: Exhaustive Search (*ES*); Random Search (*RS*); and Optimization Scheme (**S2**). As a canonical example, we consider a system with $N = 3$ users and K resources, where we vary K, with a separable cost function as in (**P1**). Here, we have specified the individual user cost functions $J_i(n_i)\ \forall\ n_i = 1, \cdots, K$ a priori, and there is no randomness. Moreover, for simplicity, we assume an identical cost structure for each user, i.e. $J_i(\cdot) = J_k(\cdot)\ \ \forall\ i, k$. Before presenting the results, we briefly describe each of the aforementioned methods:

1. *Exhaustive Search*: Here we search the entire state space $\mathcal{S}$, where $|\mathcal{S}| = \frac{(K+N-1)!}{K!(N-1)!}$
2. *Random Search*: In this approach we randomly search the state space, with the added stipulation that we do not sample previously visited states. It can be shown that, given n samples (equivalently cost evaluations), the probability that we will have sampled the optimal is given by $P_o = \left[1 - \prod_{i=1}^{n}\left(1 - \frac{1}{|\mathcal{S}|-i}\right)\right]$ Thus, for an a priori chosen P_o we can determine the complexity of the approach.
3. *Optimization Scheme* (**S2**): In this approach, given an initial $\theta^{(0)}$ and $\mathbf{r}$, we use (**S2**) to converge to the optimal θ^*. The complexity of the approach is defined as the number of iterations on θ. Finally, since this method is clearly dependent upon an initial starting state (i.e., an initial $\mathbf{r}$), for comparative purposes we average the cost evaluations over 50 independent replications (i.e., different choices of $\mathbf{r}$).

Table 2 presents a comparison of these three methods. For (**S2**), the initial probabilities (for each replication) are $\theta^{(0)} = [0.7, 0.15, 0.15]$

7 Conclusions and Further Work

The optimization scheme we have presented is intended to solve combinatorially hard stochastic optimization problems. At this point, we have restricted ourselves to resource allocation problems of the type (**P1**). The analysis in this paper is limited, however,

N	K	ES	**S2**	$RS\,(P_o = 0.9)$	$RS\,(P_o = 0.5)$
3	6	28	9.96	25	14
3	9	55	9.60	49	28
3	12	91	22.3	82	46
3	15	136	22.2	122	68
3	18	190	24.8	171	95

Table 2: Comparison of Solution Techniques for a Deterministic Problem

to asserting that an optimal solution can be attained on a sample path by sample path basis. It is obviously desirable to extend Theorem 2 to an optimal solution of (P1). It is also clear that an optimal solution may be reached before the auxiliary problem sequence converges. Therefore, it is essential to determine "stopping conditions" guaranteeing that the optimal solution is reached in practical implementations of this approach.

Although it is clear that our approach exploits the structure of resource allocation problems to obtain a solution, we believe this approach may be more general. First, the gradient estimator derived in section 4 is quite general and the extension to the case where assumption *A1* is relaxed is straightforward. Second, if local minima exist it may still be possible to determine the global minimum by averaging over different randomization vectors **r**. Clearly, exploring the generality of the proposed scheme requires extensive further research.

References

[1] P. Bremaud and F.J. Vazquez-Abad, 'On the Pathwise Computation of Derivatives with Respect to the Rate of a Point Process: The Phantom RPA Method,' *Queueing Systems*, Vol. 10, pp. 249-270, 1992.

[2] C.G. Cassandras and V. Julka, 'Marked/Phantom Algorithms for a Class of Scheduling Problems', *Proc. of 31st IEEE CDC*, pp. 3215-3220, 1992 (also submitted for publication to *Queueing Systems*)

[3] C.G. Cassandras and V. Julka, 'Optimal Scheduling in Systems with Delay-Sensitive Traffic', *to appear in Proc. of 32st IEEE CDC*, 1993

[4] W.B. Gong and H. Schulzrinne, 'Application of Smoothed Perturbation Analysis to Probabilistic Routing,' *Mathematics and Computers in Simulation*, Vol. 34, pp. 467-485,1992

[5] W.B. Gong, Y.C. Ho and W. Zhai, 'Stochastic Comparison Algorithm for Discrete Optimization with Estimation', *Proc. of 31st IEEE CDC*, 1992.

[6] J. Kiefer and J. Wolfowitz, 'Stochastic Estimation of the Maximum of a Regression Function ', *Annals of Mathematical Statistics*, 23, pp. 462-466, 1952.

[7] H. Robbins and S. Monro, 'A Stochastic Approximation Method', *Annals of Mathematical Statistics*, 22, pp. 400-407, 1951.

[8] J.E. Wieselthier, C.M Barnhart and A. Ephremides, ' Optimal Admission Control in Circuit-Switched Multihop Radio Networks', *Proc. of 31st IEEE CDC*, pp. 1011-1013, 1992

[9] D. Yan and H. Mukai, 'Stochastic Discrete Optimization', *SIAM Journal on Control and Optimization*, 30, May 1992.

PID-like Control of Discrete Event Systems Having a Discrete State Model Representation

Ray DeCarlo
Purdue University
School of Elec. Eng.
West Lafayette, IN 47907

Sergey Drakunov, Kevin Passino, and Umit Ozguner
The Ohio State University
Department of Electrical Engineering
Columbus, OH 43210

Abstract

This paper considers a subclass of discrete event systems having a discrete state model representation. By way of two examples a regulator problem is formulated and solved using a PID-like control. The PID-like control is an adaptation of the ordinary discrete time PID control to systems having possibly finite input alphabets. Because of the discrete state model and possibly finite input alphabet, regulation is achieved to a bounded set in that the trajectories are uniformly ultimately bounded. A Lyapunov stability is presented which allows us to establish a connection to variable structure control theory.

Keywords: Discrete Event Systems, Discrete State Models, PID Control, Stabilization, Lyapunov Stability, Manufacturing Systems, Variable Structure Control

1. Introduction

The idea and success of PID control for discrete time and continuous time systems, suggests the possibility of formulating and analyzing similar structures for certain discrete event systems (DES). PID control relies on the concept of generating an "error signal" and taking appropriate operations (e.g., integration, differentiation) and summing the results. Hence, to achieve PID control one needs to at least have an ability to perform, e.g., sums and differences. In a general DES theoretic framework it is not always possible to perform such operations as the variables in question are often symbols. In this paper we study a class of DES where we can easily achieve something that is analogous to PID control by assuming that we have a DES that is defined on a metric space (to be specified below) and input and output alphabets defined on the integers. Our primary objective will be to design a PID-like controller that can in some sense stabilize the system (in this initial investigation we only seek to achieve a type of boundedness).

In Section 2 we provide a DES model and the problem statement and in Section 3 we study two examples which represent a first step in the modeling and design of PID-like control strategies for such systems. The particular examples involve simple manufacturing systems (one application area for DES) where we can control the amount of parts that enter a machine and sense the number of parts that the machine has put into its output buffer. We seek to design PID-like controllers that can regulate the height of the output buffer given an unknown but constant demand by a customer.

In relation to previous work, rather than taking the conventional approach to scheduling manufacturing systems [1] we take a control-theoretic perspective more along the lines of [2] where we investigate the possibility of stabilizing simple machines using PID-like control strategies. Stabilization criteria are developed using a Lyapunov approach similar to variable structure control [15] but adapted to discrete state models similar to investigations in [3] for hybrid systems. The stability we achieve is related to the stability concepts for discrete event systems introduced in [4,5,6] and the automata-theoretic concepts of how infinite strings are accepted by automata [7,8]. Related investigations into stability analysis of manufacturing systems include: (i) the stability (boundedness) analysis of manufacturing systems conducted in [9,10], and (ii) the stability analysis of systems that are modeled with Petri nets [11] studied in [12] which includes the analysis of a production network similar to the one studied in [13,14].

2. System Model, Problem Statement, and Stability

For the purposes of this paper, we shall assume that the plant has a discrete state model of the form

$$\begin{aligned} x_{k+1} &= f(x_k, u_k) \\ y_k &= g(x_k, u_k) \end{aligned} \tag{2.1}$$

where the plant states $x_k \in X$, the input $u_k \in U$, and the outputs $y_k \in Y$ under the conditions that (X,ρ_x), (U,ρ_u), and (Y,ρ_y) are discrete metric spaces. In this paper, X, U, and Y are (possibly improper) subsets of the set of integers. However, in general, this need not be the case. The important condition is that the points cannot be arbitrarily close, e.g., for any x and x' in X, then there exists a constant ρ_{min} such that

$$\inf_{x,x'} \rho(x,x') = \rho_{min} > 0$$

The discrete metric space assumption is satisfied for various discrete event system (DES) models such as general Petri nets [11] where the state space is composed of the set of n-tuples of natural numbers and vector DES as in [13,14,3].

The plant with control will have the form given in figure 1. Here, the output of the block labeled "control strategy", denoted by u_k^c, has a discrete alphabet determined by the output of the computer or microprocessor which implements the control strategy. For the sake of uniformity, u_k^c takes values in the discrete metric space (U^c,ρ_{u^c}). The map, $\gamma(\cdot)$,. represents a coding or decision algorithm between the alphabet of U^c and the alphabet U. For example, U could be a finite subset of the positive integers whereas U^c could be the entire set of integers. The map $\gamma(\cdot)$ could represent a quantization of the space U^c.

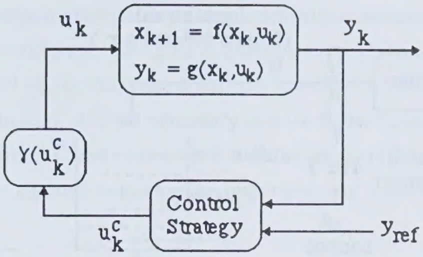


Figure 1. Plant-controller model.

The control problem will be to regulate $\rho(y_k, y_{ref})$, i.e., we want to uniformly bound $\rho(y_k, y_{ref})$ in a vicinity of the origin as k --> ∞. For the purposes of this paper we introduce the error $e_k = y_k - y_{ref}$ where all variables are defined on the integers and $\rho(y_k, y_{ref}) = | e_k |$. Hence for regulation we will require that $| e_k | \leq \varepsilon$ for some constant ε, a scalar multiple of ρ_{min}^{Y} where ρ_{min}^{Y} denotes the minimum distance in the discrete metric space Y.

In this context, the control strategies for this paper will take on two forms. The first form will be a PI-like control given by

$$u_k^c = c_p e_k + c_i z_k$$
$$u_k = \gamma(u_k^c)$$

where

$$z_{k+1} = z_k + e_k$$

represents a discrete integrator. The second controller will have the following PID-like form

$$u_k^c = c_p e_k + c_i z_k + c_d (e_{k+1} - e_k)$$
$$u_k = \gamma(u_k^c)$$

where $(e_{k+1} - e_k)$ represents the "derivative" of the error which acts as a predictor.

In the next section we will show that these control strategies can be used to regulate the output of two simple manufacturing system models.

3. Two examples

As a first example, we consider a supplier (which can be viewed as a buffer which will always have a sufficient supply of parts), a buffer (distributor of supplied goods), and a consumer model as shown in figure 2.

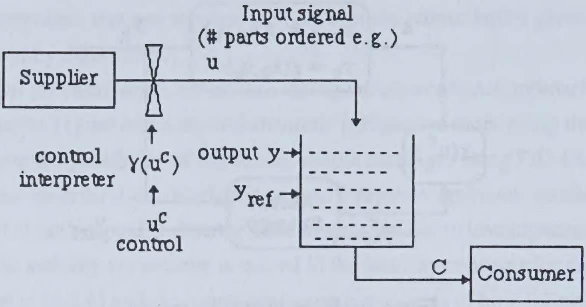


Figure 2. Simple supplier, buffer (distributor of supplied goods), consumer model.

The output variable y is the actual queue length of the buffer which takes on non-negative integer values, i.e., $y \in N$. The constant C is an integer representing the rate of consumption of parts from the buffer (i.e., the number of parts consumed at each time instant). The input variable u represents the number of supplied parts per discrete time instant. Here u takes values from a finite set $U = \{u^0, u^1, ..., u^m\}$. The control, u^c, is a signal produced by some algorithm. For our purposes, u^c takes on integer values, i.e., $u^c \in Z$. The actual input signal must be generated by an interpretation function denoted by the map $\gamma(u^c)$: Z--> N whose range is consistent with the possible positions of the actuator which control the part flow. In this example, the range of γ is a finite subset of N and $\gamma(0) = 0$; however, physically speaking it is often a finite alphabet such as {off, low, medium, high}. This could mean for example that at each time instant no parts are supplied, 5 parts are supplied, 10 parts are supplied, or 12 supplied (in this case range(γ)={0, 5, 10, 12}. It is the range "alphabet" of γ which distinguishes this framework from a standard discrete time system problem and places it in the purview of discrete event systems. For convenience however we will combine the map γ and the variable u so that the range of γ is consistent with the set U, i.e., $\gamma(u_k^c)$ will denote the number of parts supplied to the buffer at the k-th time instant. This then is consistent with the modeling formulation given in section 2.

Given this background, suppose the model of the system represented in figure 2 is

$$y_{k+1} = y_k + \gamma(u_k^c) - C \tag{3.1}$$

This model indicates that the level of "parts" in the buffer at instant k+1 equals the previous level, y_k, plus $\gamma(u_k^c)$ minus those consumed. For purposes of regulation, let y_{ref} be the (constant) desired length of the buffer queue and define the error as $e_k = y_k - y_{ref}$. The error dynamics is represented by the equation

$$e_{k+1} = e_k + \gamma(u_k^c) - C \tag{3.2}$$

Equations 3.1 and 3.2 represent discrete time integrators with constant consumption which can be viewed as a constant disturbance. Our goal is to regulate equation 3.2 and thus stabilize equation 3.1 to the constant y_{ref} in the sense described in section 2 through a discrete event interpretation of a PI controller. It is not necessary that we know C, only that C be constant.

Similar to its continuous time counterpart, a discrete event PI controller is comprised of a memoryless P element, a discrete version of an integrator, and a summation as shown in figure 3.

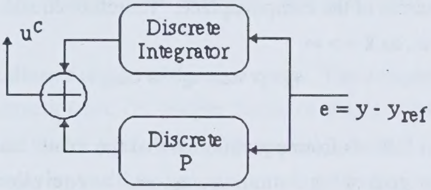


Figure 3. Block diagram of a PI-control scheme in which $u^c = c_p e_k + c_i z_k$.

The equation of the PI-controller will have the form

$$u_k^c = c_p e_k + c_i z_k \tag{3.3}$$
$$u = \gamma(u_k^c)$$

where the discrete integral part satisfies the equation

$$z_{k+1} = z_k + e_k \tag{3.4}$$

These equations are simply the ordinary discrete time PI equations restricted to the integers. The map γ defined earlier will provide an interpretation of the values of u_k^c in the context of the specific application being modeled. It is the map γ which sets this framework into the discrete event category. The controller described by equations 3.3 and 3.4 is a discrete event PI controller by the obvious fact that when $\gamma(u^c) = u^c$, then the controller is the ordinary discrete time PI controller defined on the integers.

In the case when $\gamma(u^c) = u^c$, stabilization follows by considering an auxiliary variable

$$s_k = e_k + \lambda z_k$$

where $\lambda = c_i/c_p)$. It follows from equations 3.2, 3.3, and 3.4, that

$$s_{k+1} = e_k + u_k^c - C + \lambda (z_k + e_k)$$

or equivalently

$$s_{k+1} = s_k + c_p s_k - C + \lambda e_k \tag{3.5}$$

Equations 3.5 and 3.2 taken together produce the following discrete time "equivalent" system

$$\begin{bmatrix} s_{k+1} \\ e_{k+1} \end{bmatrix} = \begin{bmatrix} 1+c_p & \lambda \\ c_p & 1 \end{bmatrix} \begin{bmatrix} s_k \\ e_k \end{bmatrix} + \begin{bmatrix} -C \\ -C \end{bmatrix} \tag{3.6}$$

By choosing c_p and c_i constrained by $\lambda = c_i/c_p$, the eigenvalues of the "A-matrix" can be assigned inside the unit circle of the complex plane. Therefore equation 3.6 will be asymptotically stable, i.e., as k --> ∞

$$s_k \dashrightarrow s_\infty, \quad e_k \dashrightarrow e_\infty$$

Under such conditions, it follows from equation 3.6 that the steady state is $s_\infty = c_i/c_p$ and $e_\infty = 0$. Hence we achieve our goal of regulation, i.e., e_k --> 0 or equivalently y_k --> y_{ref}.

This development hinges on the assumption that $\gamma(u^c) = u^c = c_p\, s_k$ wherein the control is a linear function of s_k. Typically, the map γ is a finite range quantized version of $u_k{}^c$. Hence it is necessary to consider a alternate (Lyapunov) approach which incorporates the induced structure of γ directly.

As such, define a quadratic Lyapunov function for the controller as follows

$$V_k = \begin{bmatrix} \hat{s}_k & e_k \end{bmatrix} W \begin{bmatrix} \hat{s}_k \\ e_k \end{bmatrix}, \quad W > 0 \tag{3.7}$$

where $\hat{s}_k = s_k - s_\infty$ Our objective is to find a control which leads to the inequality $V_{k+1} < V_k$ for the general system

$$x_{k+1} \equiv \begin{bmatrix} \hat{s}_{k+1} \\ e_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & \lambda \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{s}_k \\ e_k \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} \left(\gamma(u_k^c) - C\right) \equiv A\, x_k + B\left(\gamma(u_k^c) - C\right) \tag{3.8}$$

which is to be regulated in the sense described in section 2. From equations 3.7 and 3.8,

$$V_{k+1} = x_{k+1}^T\, W\, x_{k+1} = \left(A\, x_k + B\left[\gamma(u_k^c) - C\right]\right)^T W \left(A\, x_k + B\left[\gamma(u_k^c) - C\right]\right)$$

$$= x_k^T A^T W A x_k + 2B^T W A x_k\left[\gamma(u_k^c) - C\right] + B^T W B\left[\gamma(u_k^c) - C\right]^2 \tag{3.9}$$

The desired inequality $V_{k+1} - V_k < 0$ can be rewritten in the form

$$V_{k+1} - V_k = x_k^T\left(A^T W A - I\right)x_k + 2B^T W A x_k\left[\gamma(u_k^c) - C\right] + B^T W B\left[\gamma(u_k^c) - C\right]^2 < 0 \tag{3.10}$$

Since equation 3.10 is a quadratic inequality with respect to $\left[\gamma(u_k^c) - C\right]$ and $B^T W B > 0$, sufficient conditions for stability are that equation 3.10 have real roots and that $\left[\gamma(u_k^c) - C\right]$ lie between the real roots. Hence the range of γ must be sufficiently rich to allow this. The condition of real roots is met provided the quadratic of equation 3.10 have a positive value of its discriminate, i.e.,

$$\left(B^T W A x_k\right)^2 - B^T W B x_k^T \left(A^T W A - I\right) x_k > 0 \tag{3.11}$$

This inequality defines a region of the state space. The complement of this region in the state space defines the attraction set, i.e., proper choice of $\gamma(u_k^c)$, if possible, will lead to stability to the complement of the region defined by equation 3.11.

As a second example we consider incorporating a delay into the first example along with a constant disturbance d as illustrated in figure 4. The delay represents an idealization of an assembly or manufacturing process.

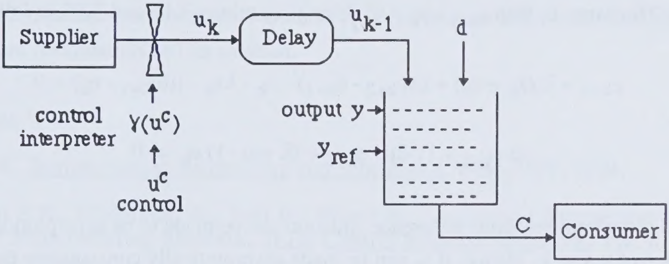


Figure 4. Extension of example 1 by inclusion of a delay element and a constant disturbance d.

Again for convenience, we will streamline the analysis by merging the total effect of $\gamma(u^c)$ and u into $\gamma(u^c)$. In this case, the mathematical model of figure 4 has the form

$$y_{k+1} = y_k + \gamma(u_{k-1}^c) - C + d \tag{3.12}$$

where d is a constant disturbance. Assuming a desired steady state of y_{ref} in the output buffer, the error dynamics to be stabilized are given by

$$e_{k+1} = e_k + \gamma(u_{k-1}^c) - C + d \tag{3.13}$$

A PID control strategy is required for the stabilization of equation 3.13. As in the analysis of the previous example, suppose for the moment that $\gamma(u_k^c) = u_k^c \in N$. Because of the delay, the control structure has the form

$$u_k^c = c_p e_k + c_i z_k + c_d(e_{k+1} - e_k) \tag{3.14}$$

where z_k satisfies equation 3.4 and represents a discrete time integrator. The term $c_d(e_{k+1} - e_k)$ represents a discrete differentiator. The causality of the differentiation is guaranteed by equation 3.13, i.e.,

$$e_{k+1} - e_k = e_k + \gamma(u_{k-1}^c) - e_{k-1} - \gamma(u_{k-2}^c) \tag{3.15}$$

which depends on past values of e_k and u_k and is independent of the disturbance. As before, we introduce the auxiliary variable

$$s_k = e_k + \lambda z_k + \mu (e_{k+1} - e_k) \tag{3.16}$$

If s_k can be made asymptotically constant, then e_k --> 0 for large k. In particular, if s_k is "constant" for large k, then $s_{k+1} - s_k = 0$. Using equations 3.15 and 3.16, it follows that

$$e_{k+1} + \lambda (z_k + e_k) + \mu(e_{k+2} - e_{k+1}) - e_k - \lambda z_k - \mu(e_{k+1} - e_k) = 0$$

Therefore,

$$\mu e_{k+2} + (1-2\mu) e_{k+1} + (\lambda + \mu - 1) e_k = 0 \tag{3.17}$$

This second order homogeneous difference equation can be made to be asymptotically stable by proper choice of μ and λ. Hence, if s_k can be made asymptotically constant, we can achieve our goal of regulation in the presence of a constant disturbance.

For the discrete event process of figure 4, the control is of course $\gamma(u^c)$ which is what distinguishes the formulation from the ordinary discrete time PID controllers. For the moment however, suppose $\gamma(u^c) = u^c$. The problem reduces to a discrete time system defined on the integers. Under this assumption, equations 3.12 through 3.16 imply that

$$s_{k+1} = (1 + \lambda) e_k + \lambda z_k + (1-\mu) (e_{k+1} - e_k) + \mu u_k^c + \mu (C-d) \tag{3.18}$$

If the PID controller of equation 3.14 is applied with

$$c_p = -\frac{1+\lambda}{\mu} + \frac{\xi}{\mu} \qquad c_i = -\frac{\lambda}{\mu} + \frac{\xi}{\mu} \qquad c_d = -\frac{1-\mu}{\mu} + \xi$$

then equation 3.18 has the form

$$s_{k+1} = \xi s_k + \mu (C-d)$$

Hence s_k --> constant provided $| \xi | < 1$.

The above stability analysis was made under the assumption that the control can take on any integer value. For the case when there is only a finite range of control values, the system can be regulated to a vicinity of the origin provided the range of $\gamma(u^c)$ is sufficiently rich. The Lyapunov stability analysis entailed here is beyond the scope of this paper, but can be carried out. However, with the γ function inserted, equation 3.18 becomes

$$s_{k+1} = (1 + \lambda)\, e_k + \lambda\, z_k + (1-\mu)\,(e_{k+1} - e_k) + \mu\, \gamma(u_k{}^c) + \mu\,(C-d) \tag{3.19}$$

It is necessary to choose $\gamma(u_k{}^c)$ so that sk approaches a constant at least on the average. This may not be possible with the limited alphabet of $\gamma(u_k{}^c)$. However, given the possibly finite range set of γ and given the values of $u_k{}^c$ from the PID control strategy, the problem is to choose $\gamma(u_k{}^c)$ so that the right side of equation 3.19 is approximately constant or constant on the average for a prescribed window of sample points. This idea is analogous to the idea of average system behavior for variable structure control systems on the sliding surface. Hence the methods of variable structure control systems [15] can be seen to have implications for this type of problem formulation and its solution.

References

[1] Baker K.R., Introduction to Sequencing and Scheduling, Wiley, NY, 1974.

[2] Gershwin S.B., Hildebrant R., Suri R., Mitter S., "A Control Perspective on Recent Trends in Manufacturing Systems", IEEE Control Systems Magazine, Vol. 6, No. 2, pp. 3-15, April 1986.

[3] Drakunov S., Dogruel M., Ozguner U., “Sliding Mode Control in Hybrid Systems”, Proc. IEEE Int. Symp. on Intelligent Control, pp. 186-189, Chicago, IL, Aug. 1993.

[4] Ozveren C.M., Willsky A.S., Antsaklis P.J., "Stability and Stabilizability of Discrete Event Dynamic Systems", MIT, LIDS Report LIDS-P-1853, Feb. 1989 (To appear in the Journal of the ACM).

[5] Ozveren C.M., Analysis and Control of Discrete Event Dynamic Systems: A State Space Approach, Ph.D. Dissertation, MIT, LIDS Report LIDS-TH-1907, Aug. 1989.

[6] Brave Y., Heymann M., "On Stabilization of Discrete Event Processes", Proceeding of the 28th Conf. on Dec. and Control, pp. 2737-2742, Tampa, FL, Dec. 1989.

[7] Buchi J.R., "On a Decision Method in Restricted Second Order Arithmetic", Proc. of the Int. Congress on Logic, Mathematics, and Phil. of Sci., 1960, Stanford Univ. Press, Stanford, CA, pp. 1-11, 1962.

[8] Muller D.E., "Infinite Sequences and Finite Machines", Switching Circuit Theory and Logic Design, Proc. Fourth Annual Symp., IEEE, Chicago, IL, pp. 3-16, 1963.

[9] Perkins J.R., Kumar P.R., "Stable, Distributed, Real-Time Scheduling of Flexible Manufacturing/Assembly/Disassembly Systems", IEEE Trans. on Automatic Control, Vol. 34, No. 2, pp. 139-148, Feb. 1989.

[10] Kumar P.R., Seidman T.I., "Dynamic Instabilities and Stabilization Methods in Distributed Real-Time Scheduling of Manufacturing Systems", IEEE Trans. on Automatic Control, Vol. 35, No. 3, pp. 289-298, March 1990.

[11] Murata T., "Petri Nets: Properties, Analysis, and Applications", Proc. of the IEEE, Vol. 77, No. 4, pp. 541-580, April 1989.

[12] Passino K.M., Michel A.N., "Stability and Boundedness Analysis of Discrete Event Systems", *Proc. of the American Control Conf.*, Chicago, IL, 3201-3205, June 1992.

[13] Li Y., Wonham W.M., "Linear Integer Programming Techniques in the Control of Vector Discrete-Event Systems", Proc. of the 27th Allerton Conf. on Comm., Control, and Computing, Univ. of Illinois at Champaign-Urbana, pp. 528-537, Sept. 1989.

[14] Li Y., Control of Vector Discrete Event Systems, Ph.D. Dissertation, University of Toronto, Dept. of Electrical Eng., May 1991.

[15] R. DeCarlo, S. Zak, and G. Matthews, "Variable Structure Control of Nonlinear Multivariable: A Tutorial," Proceedings of the IEEE, Vol. 76, No. 3, March 1988.

Optimal Sensor and Actuator Choices for Discrete Event Systems [1]

Stanley D. Young [2] and Vijay K. Garg
Department of Electrical and Computer Engineering
University of Texas at Austin, Austin, Texas 78712

Abstract

We present algorithms to optimally choose sensors and actuators to control a given discrete event system so that the closed loop behavior satisfies a specified constraint. The main results of this paper are algorithms which demonstrate: the polynomial solution to the choice of actuators, the polynomial solution to the choice of sensors for certain supervisory control problems, and the solution to the general choice of sensors.

1 Introduction

This work addresses the question of how to choose sensors and actuators to control a discrete event system to attain a specified behavior. Supervision of such systems is achieved by observing events and disabling those which are not desired or lead to undesirable behavior. A supervisor uses sensors and actuators to observe and control the system behavior.

The motivation for an optimal choice of sensors and actuators is that many systems do not require that all events be controllable or observable to attain a specified behavior. For these systems, a technique which provides a method for choosing which events to control and observe is needed. This work addresses the design question of what components a supervisor requires to control a plant to attain a specified behavior.

This work is related to other work in the optimal control of discrete event systems. In [7], optimality is defined in terms of set inclusion of the closed loop behavior of the plant and supervisor relative to the desired behavior. We require that the closed loop behavior equal the desired behavior and define optimality in terms of sensor and actuator cost. In [2], optimality is defined in terms of a cost for the sequence of events which appear in the closed loop system. In [9, 5], optimality is defined in terms of a cost for the sequence of events which appear in the closed loop system and a cost for controlling events. We assume that costs may be allocated as an initial expense and that there is no cost during system operation. We also include the costs incurred for observing events in addition to controlling them.

Previous work in the design of supervisors for discrete event systems has been of the form: given a plant, desired behavior, and a set of actuators and sensors, what is the best closed loop behavior obtainable? This work is of the form: given a plant and a desired behavior, what is the least cost set of sensors and actuators which allow the closed loop system to obtain the desired behavior?

There are two aspects to this problem: choosing the events to control and choosing the events to observe. The specific choice of events to control and observe specify the configuration of the supervisor. To determine an optimal configuration, a cost function is introduced for each event controlled or observed.

---

[1] Supported in part by NSF CCR-911065 and TRW Faculty Assistantship Award.

[2] Supported in part by a DuPont Fellowship. stanley@pine.ece.utexas.edu

For the problem considered in this paper, which requires attaining the specified behavior, the choice of which events to control is straightforward. The algorithm is based on the idea that any event which causes a behavior outside of the specified behavior must be controlled. The choice of which events to observe is not as straightforward. The algorithm proposed for the general case is exponential in the cardinality of a subset of the events which can occur in the desired behavior and polynomial in the state space of the plant and desired behavior. There are subclasses of this problem which do have a polynomial solution. Two cases in particular are considered: 1) the plant and desired behavior are represented by completely specified state machines, and 2) a "minimal" set of observed events is chosen.

In this work, the state machine formalism is used to describe the plant and constraint behaviors. We assume that the reader is familiar with the state machine formalism as given in [3] and the controllability and observability results as given in [8, 10, 1]. Section 2 introduces the problem of sensor and actuator choice and some notation. Section 3 describes a technique for choosing the events required for control. Section 4 describes a technique for choosing a potential set of events for observation. Section 4.1 describes the special case of the problem when the plant and desired behavior are represented by completely specified state machines. Section 4.2 describes the special case of the problem where the observed event set is "minimal". This approach leads to a "greedy" style of algorithm. Section 5 summarizes the results and indicates directions for future work.

2 Notation, Assumptions, and an Initial Solution

We assume that a plant and the constraint behavior are given by the languages L and K, respectively. As described in [3], with each language L and K, we associate a machine, M_L and M_K, specified by $M_L = (Q_L, A, \delta_L, q_{L,0})$ and $M_K = (Q_K, A, \delta_K, q_{K,0})$. Recall that if M is a state machine, then one writes $M = (Q, A, \delta, q_0)$ where $Q =$ a set of states, $A =$ a finite set of transition labels or events, $\delta =$ the transition function, $\delta : Q \times A \rightarrow Q$, and $q_0 =$ the initial state, $q_0 \in Q$. The machine which results from the product of M_L and M_K is used extensively. We denote this machine by $M_{||} = (Q_{||}, A, \delta_{||}, q_{||,0})$.

We assume that the primary cost associated with a sensor or an actuator is the initial capital investment and that the cost of operation is negligible compared to the initial cost. Hence, the cost functions used for evaluating optimality concern only the event to be observed, or controlled and not the number of times an event is observed or controlled [3]. The actuator and sensor cost functions are denoted by $c_c : A \rightarrow \Re^+$ and $c_o : A \rightarrow \Re^+$.

To specify that a specific event has been chosen for observation or control, "choice" functions are used. The "choice" functions used in this work are denoted by $i_c : A \rightarrow \{0, 1\}$ and $i_o : A \rightarrow \{0, 1\}$ for the control and observation "choice" functions, respectively. For these functions, $i_c(\sigma) = 0$ ($i_o(\sigma) = 0$) means that event σ has not been chosen for control (observation), and $i_c(\sigma) = 1$ ($i_o(\sigma) = 1$) means that event σ has been chosen for control (observation). Now the optimization problem can be specified.

Problem 2.1 Given a plant behavior specified by the language L, a constraint behavior specified by the language K, with $K \subseteq L$, and cost functions c_c and c_o, choose i_c and i_o to minimize the cost C such that $L_f = K$, where

$$C = \sum_{\sigma \in A} (c_c(\sigma) \cdot i_c(\sigma) + c_o(\sigma) \cdot i_o(\sigma)),$$

---

[3] Note that in the general case a state dependence for the costs may be required.

and L_f is the closed loop behavior as described in [8].

In general, functions i_o and i_c can always be found such that $L_f = K$. The solution in which everything is chosen to be controlled and observed i.e. $\forall \sigma \in A : i_c(\sigma) = i_o(\sigma) = 1$, provides that $L_f = K$. However, this solution is most likely not the minimum cost solution.

An obvious choice to determine the optimal i_c and i_o functions is to determine the cost for each possibility and pick one which has the minimal cost and satisfies $L_f = K$. It is clear that the procedure of trying each possible function for i_o and i_c can be used to find the minimum. By [10], this procedure has a complexity of $O(2^{|A|} \cdot 2^{|A|} \cdot |\tilde{Q}|^3 \cdot |A|)$, where $\tilde{Q} = \max(Q_L, Q_K)$.

3 A Calculation for the Control "Choice" Function

The problem with the approach given above is the many exponentials which appear in the complexity measure. We now consider ways to reduce some of this complexity.

First, we consider ways to calculate i_c, the control "choice" function. In order to calculate the required function, consider the product $M_{||}$ of M_L and M_K. Each state $q_{||} \in Q_{||}$ in this product machine has corresponding states $q_L \in Q_L$ and $q_K \in Q_K$.

Definition 3.1 $E_c : Q_{||} \rightarrow 2^A$ is the function which gives the set of events $\sigma \in A$ which are defined in M_L at state q_L and not defined in M_K at q_K, i.e.

$$E_c(q) = \{\sigma \in A : \delta_L(q_L, \sigma)! \wedge \neg\delta_K(q_K, \sigma)!\},$$

where $q = (q_L, q_K)$, and $\delta(q, \sigma)!$ means that σ labels a transition from state q.

Next we extend this definition to the entire state set, by defining the set

$$\hat{E}_c = \bigcup_{\{q \in R_{||}(q_{||,0})\}} E_c(q),$$

where $R_{||}(q_{||,0}) = \{q \in Q_{||} : (\exists w \in A^* : \delta_{||}(q_{||,0}, w) = q)\}$ and A^* is the set of finite length strings of symbols from alphabet A. In other words, the controllable events are those which must be disabled at states which are reachable from the initial state.

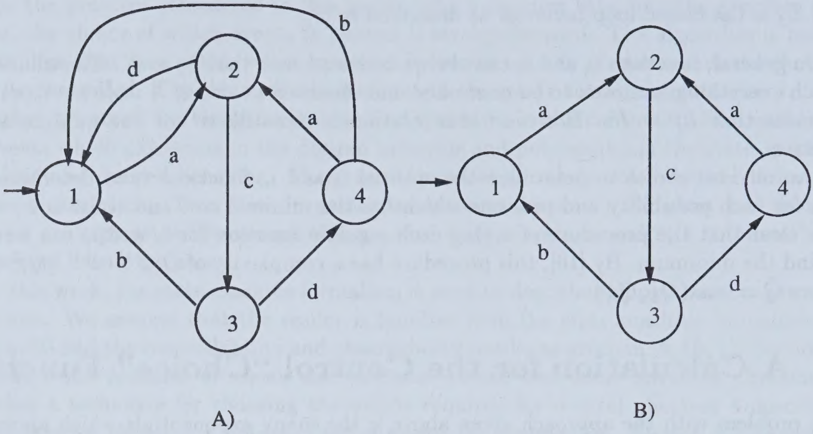
Proposition 3.1 $\hat{E}_c$ is the smallest set of controllable events such that $L_f = K$.

The proof is based on the requirement that $L_f = K$.

A result of this proposition is that we can drop the distinction between $\hat{E}_c$ and A_c, where A_c, A_{uc} are the sets of controllable and uncontrollable events, respectively [8]. Let A_c denote this minimal $\hat{E}_c$.

Since calculating the product machine has complexity $O(|Q_L| \cdot |Q_K| \cdot |A|)$, we can replace one of the exponential factors in the complexity of the procedure specified in Section 2 by $|\tilde{Q}|^2 \cdot |A|$. This reduction results from replacing the tests for i_c with the calculation of the product machine and construction of $E_c(q)$ for each q. Also, note that this calculation must be done only once. Hence, it is an additive term and is not counted as part of the observation function calculation. The resulting complexity with this modification is $O(2^{|A|} \cdot |\tilde{Q}|^3 \cdot |A|)$.

An illustration of the controlled event set calculation is given in Example 3.1.



A) B)

Figure 1: A) Plant and B) Specified Behavior.

Example 3.1 Consider the plant and constraint behaviors as given in Figure 1. For these machines, the part of the product machine which is reachable from the initial state is isomorphic to the constraint behavior. By comparing the states in these machines, it is clear that only at states 4 and 2 in the constraint machine are there any events which are defined in the plant and not in the constraint behavior. Hence, we have that $E_c(4) = \{b\}$, $E_c(2) = \{d\}$, and for all other states $E_c(q) = \emptyset$. For this system, the events which must be controllable are b and d, i.e. $A_c = \{b, d\}$.

The final result in Proposition 3.2 shows that the method prescribed has optimal computational complexity.

Proposition 3.2 The complexity of calculating the events which must be controlled to obtain the specified behavior is $\Theta(|\tilde{Q}|^2 \cdot |A|)$.

The proof of this proposition is obvious by considering that calculating the controllability of a language with respect to another one is a special case of the problem considered here. This controllability calculation is known to be $\Theta(|\tilde{Q}|^2 \cdot |A|)$ [10].

4 Calculation of the Observation "Choice" Function

Now we consider ways to calculate i_o, the observation "choice" function.

To obtain the correct closed loop behavior, $L_f = K$, a supervisor needs to identify which control pattern to use for the current state. Recall that a control pattern which a supervisor applies at a given state may be interpreted as a bit pattern with one bit for each event. A 1 in an event's position means that it is enabled, a 0 means that it is disabled. Thus, any event σ labeling a transition between states [4] q_1 and q_2, which require sufficiently different control patterns, needs to be observed. Now we must investigate what conditions constitute a sufficient difference.

---

[4] Notice that in the remaining discussion most of the references to states and machines concern the product machine.

Let γ_q be the control pattern for state q and $E_c(q)$ be the events which must be disabled at state q. Let the function $E_d : Q \rightarrow 2^A$ give the transitions which are defined from state q, i.e.

$$E_d(q) = \{\sigma \in A : \delta_{||}(q, \sigma)!\}.$$

There are certain types of transitions which require a change in control patterns and hence must be observed. Consider a transition from state q_1 to q_2 by event σ, i.e. $\delta_{||}(q_1, \sigma) = q_2$. The transition for which $E_c(q_1) \cap E_d(q_2) \neq \emptyset$ must be identified. In this case, an event which must be disabled at q_1 is defined for q_2. Also, the transition for which $E_d(q_1) \cap E_c(q_2) \neq \emptyset$ must also be identified. In this case, an event which is defined at q_1 must be disabled at q_2.

Definition 4.1 A *Type O* transition is defined by

$$(E_c(q_1) \cap E_d(q_2) \neq \emptyset) \vee (E_d(q_1) \cap E_c(q_2) \neq \emptyset).$$

Other transitions might or might not require observation, depending on what control patterns the supervisor chooses for the states associated with the transition, as shown in Example 4.1.

Example 4.1 If none of the sets describing the controlled events and the defined events intersect, then the supervisor can assign the control pattern for each of the states to be the same. I.e. if $E_c(q_1) \cap E_c(q_2) = \emptyset$, $E_c(q_1) \cap E_d(q_2) = \emptyset$, $E_d(q_1) \cap E_c(q_2) = \emptyset$, and $E_d(q_1) \cap E_d(q_2) = \emptyset$, then the supervisor can set $\gamma_{q_1} = \gamma_{q_2} = E_c(q_1) \cup E_c(q_2)$ and need not identify this transition for local control purposes. A specific illustration of these circumstances can be seen in states $(1, 1)$ and $(2, 2)$ of the product of the machines given in Figure 1.

Proposition 4.1 Given machines M_K and M_L, to obtain $L_f = K$, we have the following condition on the observed events:

if

$$\exists q_1, q_2 \in Q_{||} : (q_1 \in R_{||}(q_{||,0})), \sigma \text{ is } \textit{Type O}, \text{ and } \delta_{||}(q_1, \sigma) = q_2,$$

then

$$\sigma \in A_o,$$

where $R_{||}(q_{||,0})$ is as defined in Section 3 and A_o, A_{uo} are the sets of observed and unobserved events, respectively [8].

The proof of this proposition is obvious and follows from the requirement that $L_f = K$ and that q_1 is reachable from the initial state.

Let the events which satisfy the conditions in Proposition 4.1 form a set denoted by A_{req} (for the required events).

The procedure for choosing sensors and actuators which results from the above considerations consists of the following steps: calculation of the events which must be controlled, A_c, calculation of the events which must be observed, A_{req}, and then trying all choices for the observation "choice" function which are consistent with A_{req}. The reduction in complexity arises from the reduction in the average number of potential cases which must be considered for A_o. Since calculating $M_{||}$, A_c, and A_{req} have complexity $O(|\tilde{Q}|^2 \cdot |A|)$, and by [10] checking controllability and observability is $O(|\tilde{Q}|^3 \cdot |A|)$, we obtain a complexity of $O(2^{|A \setminus A_{req}|} \cdot |\tilde{Q}|^3 \cdot |A|)$ for this approach.

4.1 Fully Specified Behaviors

We consider some special cases as a means of further reducing the complexity of choosing the required sensors and actuators. The special case considered in this section is based on the polynomial complexity of finding the reduced machine for a fully specified Moore machine. The approach is to form the product of the plant and desired behavior and then to finish specifying the machine such that it is a fully specified Moore machine [4, 3], where the output at a state represents a bit pattern corresponding to the control pattern at the state. In addition to an output control pattern for each state, a fully specified machine has a transition specified for every event from every state. In general, most machines used to represent discrete event systems are not fully specified, i.e. there is usually not a transition specified for every event from every state. This characteristic arises from the fact that in the actual system it may not make any sense to define a transition for each event from every state.

By [4], calculating the minimal equivalent state machine is polynomial in the number of states, i.e. $O(|\tilde{Q}|^2 \cdot |A|)$. Hence, if we can reduce the general problem of observed event set selection to the problem of determining the minimal equivalent machine, then we will have a polynomial technique for choosing the controlled and observed events. Recall that for an incompletely specified Moore machine determining the minimal equivalent machine has nondeterministic polynomial time complexity and is complete in this class [6]; consequently, we consider this approach only for completely specified machines. Note that in general this approach will not provide an optimal choice for the observed event set, due to possible suboptimal completions of the machine specification.

There are two aspects to completing the specification of the product machine: specifying the transitions and the output patterns.

The transitions may be specified in any manner consistent with the original product machine such that the resulting machine is deterministic.

From the incompletely specified product machine, one can calculate the events which must be controlled and those which can remain uncontrolled. This information is used to fully specify the output patterns. Recall that an uncontrollable event cannot be disabled by a supervisor. This fact provides a constraint on specifying the output patterns: any uncontrollable event must be enabled in every output pattern. There are two ways to specify the output pattern for the controlled events. Any controllable event which is not defined in the original product machine at a given state may be disabled in the output pattern for that state. Alternatively, the event may be enabled or disabled in an attempt to provide a lower cost solution. (This second alternative essentially requires the specifier to make the choices which place the incompletely specified problem in the nondeterministic polynomial complete complexity class.)

Observe that the first technique of choosing the output patterns has a benefit beyond providing a unique and deterministic technique for choosing the observed event set. The supervisor which results from this choice may be used with a wide collection of plants and still guarantee the correct closed loop behavior.

Example 4.2 Consider the original plant and specified behavior as given in Figure 2.

The product machine is isomorphic to the machine for the desired behavior. It is clear that $A_c = \{a, c\}$ and $A_{uc} = \{b\}$. Let λ denote the output function for this machine. From the first alternative for selecting the output patterns, we obtain that $\lambda(1) = 110$, $\lambda(2) = 010$, and $\lambda(3) = 011$. This obviously is not optimal because to generate the correct output both a and b must be observed, whereas it is sufficient to observe only

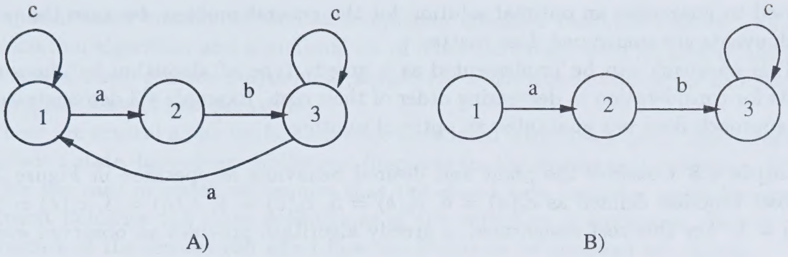


Figure 2: Plant and Specified Behavior for a Simple System.

a or b. The benefit of this approach is apparent when one considers how many other plants, with the same state space and event set, can be controlled with the supervisor which results from this case. There are 4^4 possible plants which would give the same result. Hence, this approach provides a supervisor which can be applied in a wide variety of cases in addition to the original plant.

The justification for considering events which label transitions in the minimal equivalent Moore machine is given in the Proposition 4.2.

Proposition 4.2 For a given completely specified Moore machine representation of the product of M_L and M_K, an event σ is in the observed event set, i.e. $\sigma \in A_o$, if and only if σ labels a transition between distinct states in the minimal equivalent Moore machine.

The proof of this proposition is based on the requirement that the output patterns in the minimal machine match those in the original machine on each input string.

4.2 "Minimal" Choice of Observed Events

The special case considered in this section gives a minimal versus an optimal solution. The "minimal" choice of observed events is obtained by a greedy algorithm based on the observation cost function.

Definition 4.2 Let K and L denote the desired and plant behaviors, respectively. Let $P(B)$ denote the mask associated with the observed event set B. We say that B is *minimal* if there is no set X such that $X \subseteq B$ and $B \neq X$ with K observable with respect to L and $P(X)$.

In general, there will be several minimal sets of observed events.

This approach calculates the minimal set of observed events based on the order in which events are chosen to determine if an event is required for observability. Note that this choice is what provides the polynomial complexity of this approach.

The technique consists of choosing an event and determining if the desired behavior is observable with respect to the plant behavior and the proposed set of observed events. The proposed set of observed events is the current set of observed events minus the chosen event. If the behavior is observable, then remove this event from the set of observed events. If it is not, then pick another event to test. Continue until all events have been tested. The complexity of this procedure is $O(|A| \cdot |\tilde{Q}|^3)$. Note that this procedure cannot be used to guarantee an optimal solution for the general problem because the order in which events are considered does matter.

This approach can be implemented as a greedy type of algorithm by choosing the events for consideration in decreasing order of their cost. Example 4.3 demonstrates that this approach does not guarantee an optimal solution.

Example 4.3 Consider the plant and desired behaviors as specified in Figure 3 with the cost function defined as $c_o(a) = 6$, $c_o(b) = 5$, $c_o(c) = 4$, $c_o(d) = 3$, $c_o(e) = 2$, and $c_o(f) = 1$. For this cost assignment, a greedy algorithm provides an observed event set

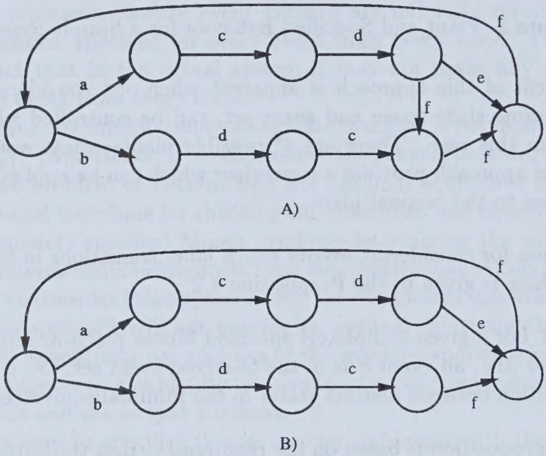


Figure 3: A) Plant and B) Specified Behavior.

$A_o = \{c, d, e\}$ with an observation cost of 9 units. An optimal observed event set is $A_o = \{a, e\}$ with an observation cost of 8 units.

5 Conclusions

This paper addresses the issue of how to choose sensors and actuators for a system which may be modeled using state machines. The goal is to restrict the system's behavior to a specified constraint behavior. To insure that the system adheres to the constraint behavior certain events in the system must be observed and controlled with sensors and actuators.

We have provided a technique for determining which events must be observed and controlled to achieve the specified constraint behavior. A cost function associated with the sensors and actuators is used to provide a metric for choosing between different controller configurations.

Using different types of equivalence classes, an algorithm is given for calculating the controller and observer parts of the supervisor which restricts the system's behavior to the specified constraint behavior. The algorithm for choosing the actuators has polynomial complexity in terms of the events and the states in the product machine. The approach presented in this work provides an algorithm for choosing sensors and actuators with exponential complexity in the number of events and polynomial complexity in the states.

The special cases considered were a completely specified machine with its associated minimization algorithm and a minimal set of observed events with its greedy algorithm. Both special cases provide polynomial complexity in the number of events and states for choosing the sensors and actuators.

There are several areas for further work associated with this paper. One extension is to include a state dependence of the cost functions for the sensors and actuators. Observe that for the case in which we require that the closed loop behavior equal the specified constraint behavior the state dependence of the actuators is irrelevant. But the state dependence of the sensors can affect how the events to be observed are chosen.

Another extension is to allow the closed loop behavior to be a subset of the specified constraint behavior. In this case, a technique for measuring how much of the specified behavior is not attained is needed. This measure would need to have an associated cost and be included in the minimization Problem 2.1. One approach assigns a cost to states and uses the reachability of the state in the closed loop behavior as a measure of how much of the specified constraint behavior is attained. This approach is similar to that specified in [5]. Another approach is to assign a measure to the strings in the specified constraint behavior which are not possible in the closed loop behavior and then to assign a cost to this measure and include it in the minimization problem. In either of these cases, the state dependence of the cost for sensors and actuators must be included in the general case analysis of the minimization problem.

References

[1] R.D. Brandt, V.K. Garg, R. Kumar, F. Lin, S.I. Marcus, and W.M. Wonham. "Formulas for calculating supremal controllable and normal sublanguages". *Systems & Control Letters*, 15:111–117, 1990.

[2] Y. Brave and M. Heymann. "On Stabilization of Discrete Event Processes". *Internation Journal Control*, 51:1101–1117, 1990.

[3] J.E. Hopcroft and J.D. Ullman. *Introduction to Automata Theory, Languages and Computation.* Addison-Wesley, Reading, MA, 1979.

[4] Z. Kohavi. *Switching and Finite Automata Theory.* McGraw-Hill, New York, NY, 1978.

[5] R. Kumar and V.K. Garg. "Optimal control of discrete event dynamical systems using network flow techniques". In *Proceedings 1991 Annual Allerton Conference*, pages 705–714, Urbana, IL, October 1991.

[6] C. P. Pfleeger. "State Reduction in incompletely specified finite state machines". *IEEE Transactions Computers*, 22(12):1099–1102, 1973.

[7] P.J.G. Ramadge and W.M. Wonham. "Supervisory control of a class of discrete event processes". *SIAM J. Control Optimization*, 25:206–230, January 1987.

[8] P.J.G. Ramadge and W.M. Wonham. "The control of discrete event systems". *Proceedings of IEEE*, 77(1):81–98, January 1989.

[9] R. Sengupta and S. Lafortune. "Optimal control of a class of discrete events systems". In *Proceedings 1991 IFAC symposium on distributed intelligence systems*, pages 25–30, Arlington, VA, August 1991.

[10] J.N. Tsitsiklis. "On the control of discrete event dynamical systems". *Math. Contr., Signals, and Syst.*, 2(1):95–107, 1989.

Control Synthesis Using Path Algebras for Forbidden State Avoidance

Lawrence E. Holloway*
Department of Electrical Engineering and
Center for Robotics and Manufacturing Systems
University of Kentucky
Lexington, Kentucky 40506

Abstract

For a general class of controlled Petri nets (CtlPNs), control laws for the avoidance of forbidden states can be formulated using path algebra expressions. These expressions are determined through analysis of the net structure, and compactly represent reachability conditions for forbidden states. This paper presents a feedback control method for using the path algebra expressions to determine a control. The method is divided into an off-line analysis of the net structure which determines the algebraic expressions, and then an on-line evaluation of the expressions to determine the appropriate control.

1 Introduction

Discrete event control theory considers the synthesis of control policies for discrete event systems (DES). Several formalisms have been used for modeling discrete event plants, including formal languages, automata, and recursive algebras, among others [1, 2, 3, 4]. One formalism which has received recent attention is controlled Petri nets [5, 6]. Li and Wonham have extended the notion of DES controllability and observability to controlled Petri nets [7]. Ichikawa and Hiraishi considered control for directing the system to a target state [5]. Holloway and Krogh have developed methods for avoiding forbidden states in marked graph Petri nets, and later extended this method to guarantee liveness of the control [8, 9, 10]. Ben-Naoum, Van Breusegem, and Boel consider forbidden state avoidance in automata Petri nets [11]. Event sequencing control specifications and state sequencing specifications are addressed in [12, 13, 7]. Timing control is examined in [14, 15].

In this paper we consider the formulation of a feedback control policy for avoiding forbidden states. The controlled Petri nets we consider are capable of modeling process synchronization typical of marked graph Petri nets, and process choice as found in automata Petri nets. This work is thus a generalization of previous works which examined exclusively either marked graph nets or state machine nets [8, 11]. This paper should be viewed as a companion paper to Holloway and Guan [16]. In [16], the theoretical basis of the method is established in a rigorous manner. In this paper, the use of the method is described in a step-by-step manner. Detailed definitions and a more analytical discussion are found in [16].

The forbidden state avoidance control problem can be stated as follows: Given a set of forbidden states (Petri net markings) $M_{\mathcal{F}}$, and given a current state m, determine a control that will prevent the system from reaching any forbidden state. The control formulation is

---

*This work is supported in part by NSF grant ECS-9308737 and the Center for Robotics and Manufacturing Systems, University of Kentucky.

complicated by the presence of uncontrolled transitions in the net, so that a forbidden marking may be reached through some uncontrollable sequence of state changes (transition firings). The set of markings that can uncontrollably lead to a forbidden marking is called the set of *weakly forbidden markings*, denoted $W(M_{\mathcal{F}})$. The set of markings which are not weakly forbidden but which can lead to a weakly forbidden marking within a single state change without the appropriate control are called *boundary markings*, and are denoted $\delta(W(M_{\mathcal{F}}))$.

It is important to identify the boundary markings to develop a control policy. However, the state space of a controlled Petri net can be very large, so it is often infeasible to enumerate all markings and individually label them as weakly forbidden markings, boundary markings, or neither. In this paper and [16], we describe a method of characterizing the reachability of forbidden states as algebraic expressions. These expressions are composed of terms representing paths in the net, and are called *path algebra* expressions. Evaluating an expression for a given marking will determine whether the marking is weakly forbidden or whether it is a boundary marking. This evaluation will also indicate the controls which must be imposed on the net. The control policy we present is divided into two parts. The off-line analysis develops the expressions characterizing the reachable marking space, and the on-line part determines the set of allowable controls by evaluating the algebraic expressions for the current marking.

The following section reviews controlled Petri nets and forbidden state specifications. Section 3 describes the off-line development of the algebraic expressions characterizing the net structure. Section 4 describes the on-line control determination, and section 5 summarizes the analytical results of [16]. Section 6 summarizes the paper.

2 Controlled Petri Nets

Controlled Petri nets have been defined in several papers, including [16, 8, 9], so only a brief description is given here. A controlled Petri net (CtlPN) is a tuple $\mathcal{G} = (\mathcal{P}, \mathcal{T}, \mathcal{E}, \mathcal{C}, \mathcal{B})$, where $\mathcal{P}$ is a finite set of *state places*, $\mathcal{T}$ is a finite set of *transitions*, $\mathcal{E} \subseteq (\mathcal{P} \times \mathcal{T}) \cup (\mathcal{T} \times \mathcal{P})$ is a set of directed arcs connecting state places and transitions, $\mathcal{C}$ is the finite set of control places, and $\mathcal{B} \subseteq (\mathcal{C} \times \mathcal{T})$ is the set of directed arcs associating control places with transitions. The notation ${}^{(p)}t$ and $t^{(p)}$ indicate the set of input and output state places, respectively, for the transition t. Similarly, ${}^{(t)}p$ and $p^{(t)}$ are respectively the input and output transitions for state place p.

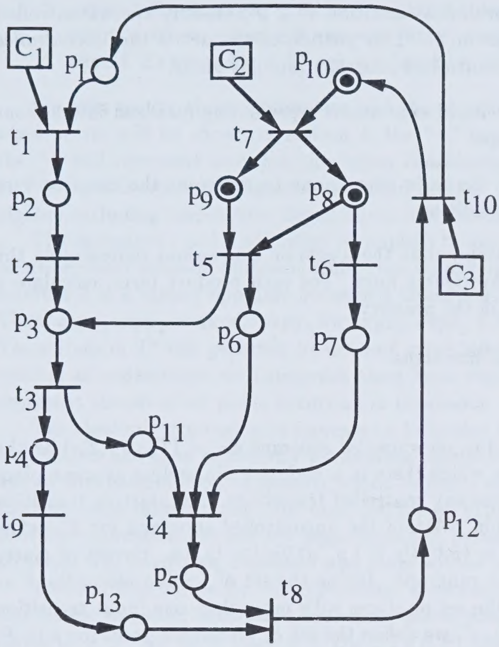
A *path* $\pi \in (\{tp \mid p \in \mathcal{P}, t \in \mathcal{T}\})^*$ is an alternating sequence of transitions and places representing a directed path in the CtlPN structure. Paths considered in this paper will always begin with a transition and end with a place. The starting transition for a path π is denoted t_π.

The *marking* (state) of a CtlPN is a function which associates a non-negative integer to each state place in the net. The integer represents the number of *tokens* in the place. The set of all markings is denoted $\mathcal{M}$. Figure 1 illustrates a controlled Petri net, where state places are indicated by circles, transitions by bars, and control places by squares. Tokens are represented by black dots in the state places.

A *control* $u : \mathcal{C} \longrightarrow \{0, 1\}$ assigns a binary token count to each control place in the net.The set of all controls is denoted $\mathcal{U}$. Given two controls $u, u' \in \mathcal{U}$, control u' is said to be *more permissive* than u if $u'(c) \geq u(c)$ for all $c \in \mathcal{C}$ and $u'(c) > u(c)$ for at least one $c \in \mathcal{C}$. A (nondeterministic) *feedback control law* is a function $\mathcal{U} : \mathcal{M} \longrightarrow 2^{\mathcal{U}}$ which assigns to each marking a set of possible controls.

A set of transitions $T \subseteq \mathcal{T}$ is *state enabled* under a marking m when $m(p) \geq \mid p^{(t)} \cap T \mid$ for all $p \in {}^{(p)}t$. A set of transitions T is *enabled* under a marking m and a control u if it is both state enabled and $u(c) = 1$ for all $c \in {}^{(c)}t$ for all $t \in T$. A state change occurs when a set of enabled transitions $T \subseteq \mathcal{T}$ *fires*, which results in a new marking m' defined by the following equation:

$$m'(p) = m(p) - \mid p^{(t)} \cap T \mid + \mid {}^{(t)}p \cap T \mid$$



$$\pi_1 = (t_1 p_2 t_2 p_3)$$
$$\pi_2 = (t_7 p_9 t_5 p_3)$$
$$\pi_3 = (t_7 p_8)$$
$$\pi_4 = (t_5 p_3)$$
$$\pi_5 = (t_3 p_4)$$
$$\pi_6 = (t_5 p_6 t_4 p_5)$$
$$\pi_7 = (t_6 p_7 t_4 p_5)$$
$$\pi_8 = (t_7 p_9 t_5 p_6 t_4 p_5)$$
$$\pi_9 = (t_3 p_{11} t_4 p_5)$$

Figure 1: A controlled Petri net and the path set Π_F for $F = \{p_4, p_5\}$

Thus, the marking of each state place is decremented by the firing of an output transition of the state place and incremented by the firing of an input transition to the state place.

Given a marking m and a control u, a marking m' is said to be *immediately reachable* if there exists a set of transitions T such that T is enabled under m and u, and the firing of T results in the marking m'. The set of immediately reachable markings under a marking m and a control u is denoted $R_1(m,u)$. A marking m_k is said to be *reachable* from a marking m_0 under a control $u \in \mathcal{U}$ if there exists a sequence $(m_0 m_1 \ldots m_k)$ such that $m_i \in R_1(m_{i-1}, u)$ for $0 < i \leq k$. The set of all reachable markings from m under u is denoted $R_\infty(m,u)$.

The control goal considered in this paper is expressed in terms of *forbidden conditions.* A forbidden condition is a set of places $F \subseteq \mathcal{P}$. The set of markings which *satisfy* a forbidden condition F is defined as M_F, where $m \in M_F$ if $m(p) \geq 1$ for each place $p \in F$. Given a set of forbidden conditions $\mathcal{F}$, the set $M_\mathcal{F}$ of forbidden markings for $\mathcal{F}$ is the union of M_F for all $F \in \mathcal{F}$. Our control goal is to prevent the system from reaching any forbidden marking $m \in M_\mathcal{F}$. We define the weakly forbidden marking set as $W(M_\mathcal{F}) \equiv \{m \in \mathcal{M} \mid R_\infty(m,u) \cap M_\mathcal{F} \neq \emptyset$ for some $u \in \mathcal{U}\}$. The set of boundary markings are defined as $\delta(W(M_\mathcal{F})) \equiv \{m \in \mathcal{M} - W(M_\mathcal{F}) \mid R_1(m,u) \cap W(M_\mathcal{F}) \neq \emptyset$ for some $u \in \mathcal{U}\}$.

3 Off-Line Analysis

The control method in this paper is divided into an off-line analysis and an on-line control determination. In this section, we consider the off-line steps. The off-line analysis for the net is independent of the current marking. The purpose of the off-line analysis is to develop algebraic expressions which can be evaluated on-line to determine if and when controls must be disabled. The off-line computation is divided into four steps:

1. **Path identification:** For each forbidden condition $F \in \mathcal{F}$, identify the uncontrolled portion of the net leading to places in F. This portion of the net is the *uncontrolled subgraph* for F. Decompose this uncontrolled subgraph into net paths.
2. **Basic expression construction:** Build expressions representing localized interactions of paths in the uncontrolled subgraph.
3. **Expression expansion:** Expand the basic expressions to represent the net structure leading to places in the forbidden condition.
4. **Token threshold calculation:** Manipulate the algebraic expressions representing the forbidden conditions into a sum-of-products form. For each product term, calculate a *token threshold* for each path term in the product.

We now examine each of the above off-line steps.

Path Identification

For a forbidden condition F, define the *uncontrolled subgraph* $\mathcal{G}_F = (\mathcal{P}_F, \mathcal{T}_F, \mathcal{E}_F)$ as the set of all places, transitions, and arcs for which there is a path $\pi \in \Pi_{\mathcal{G}}$ ending at some place $p \in F$, such that the path does not traverse any controlled transitions. The starting transition of the path may be a controlled transition. Within the uncontrolled subgraph for F, define the set of *branching places* for F as $D_F \equiv \{p \in \mathcal{P}_F \mid \ | p^{(t)} \cap \mathcal{T}_F \,| > 1\}$, i.e. the set of places with more than one exit transition in the subgraph. Define the set of *joining places* for F as $J_F \equiv \{p \in \mathcal{P}_F \mid \ | {}^{(t)}p \cap \mathcal{T}_F \,| > 1\}$, i.e. the set of places with more than one input transition in the subgraph. For each $p \in D_F \cup J_F \cup F$, we define the set of *precedence paths* for p in F, $\Pi_F(p) \subseteq \Pi_{\mathcal{G}}$, as the set of all paths $\pi \equiv (t_1 p_1 \ldots t_n p_n)$ in $\mathcal{G}_F$ such that

1. The path terminates at the place p;
2. The starting transition of the path is either a controlled transition or has as input a place in $(D_F \cup J_F \cup F)$;
3. The path does not overlap itself, i.e. $p_i \neq p_j$ and $t_i \neq t_j$ for $1 \leq i \neq j \leq n$;
4. No place other than the terminating place p can be an element of $D_F \cup J_F \cup F$.

Define Π_F as the union of $\Pi_F(p)$ for all $p \in D_F \cup J_F \cup F$. Note that it is possible for two paths $\pi \in \Pi_F(p)$ to overlap each other – as long as they join or branch from a transition instead of a place. It is even possible for one path to be contained within the other, as long as the transition t_π for the contained path has ${}^{(p)}t_\pi \cap (D_F \cup J_F \cup F) \neq \emptyset$.
Example: For the net and forbidden condition $F = \{p_4, p_5\}$ in figure 1, $D_F = \{p_8\}$ and $J_F = \{p_3\}$. The paths in Π_F are shown alongside the net in figure 1.

Basic Expression Construction

The paths terminating in the same place indicate possible token travel routes that could lead to a marking of the place. The relationships between these paths are indicated by a *basic path expression.* However, before describing the construction of these basic path expressions, we must characterize the algebra of the expressions that we consider. The algebra consists of three operators, "+", "$\cdot$", and "$\diamond$". The set of expressions in the algebra is then defined as follows:

Definition 1 : The set EXPR of expressions in the algebra is defined recursively using the three binary operators $+$, $\cdot$, and $\diamond$:

1. $\pi \in \Pi_{\mathcal{G}} \Longrightarrow \pi \in \text{EXPR}$

2. $\mathtt{expr}_1 \in \textsc{Expr}, \mathtt{expr}_2 \in \textsc{Expr} \Longrightarrow (\mathtt{expr}_1 + \mathtt{expr}_2) \in \textsc{Expr}$
3. $\mathtt{expr}_1 \in \textsc{Expr}, \mathtt{expr}_2 \in \textsc{Expr} \Longrightarrow (\mathtt{expr}_1 \cdot \mathtt{expr}_2) \in \textsc{Expr}$
4. $\mathtt{expr} \in \textsc{Expr}$ and $\sigma \in \mathcal{T}^* \Longrightarrow \sigma \diamond (\mathtt{expr}) \in \textsc{Expr}$.

We refer to the above expressions and the associated properties discussed in [16] as a *path algebra.* As will be shown in section 4, the "+" operator will be evaluated as a disjunction, the "·" will represent a conjunction when considering expressions with disjoint sets of paths, and "◇" will indicate a designated travel for a token in a given path. Defined properties of the algebra including associative, distributive, commutative, and identity are given in [16].

The operators · and ◇ will often be implicit in an expression. The implicit operators should be understood without confusion since · is a binary operator between two expressions in $\textsc{Expr}$, whereas ◇ is a binary operator between a transition string in $\mathcal{T}^*$ and an expression in $\textsc{Expr}$. Thus, $\mathtt{expr}_1\mathtt{expr}_2 \equiv \mathtt{expr}_1 \cdot \mathtt{expr}_2$ for $\mathtt{expr}_1, \mathtt{expr}_2 \in \textsc{Expr}$, and ${}^{\sigma}\mathtt{expr}_1 \equiv \sigma \diamond \mathtt{expr}_1$ for $\sigma \in \mathcal{T}^*$. The strings in $\mathcal{T}^*$ will generally be denoted using indexed variables on the symbol σ and will be written as superscripts to distinguish them from expressions. Let the notation $\Pi(\mathtt{expr}) \subseteq \Pi_{\mathcal{G}}$ represent the set of all paths occurring in expression $\mathtt{expr}$.

The algebraic expression to represent a forbidden condition is constructed by building basic expressions and then recursively performing substitutions on the terms of the expressions. We use an overbar notation to identify which terms of an expression have not yet been expanded through term substitutions. For example, for an expression $\mathtt{expr}$, the notation $\overline{\mathtt{expr}}$ indicates that the expression has not yet been expanded by any substitutions. In the final expressions, all terms will have been expanded and there will be no overbar terms.

The building blocks for our expressions are unexpanded expressions for places in forbidden conditions and their associated branching places and joining places. For each forbidden condition $F \in \mathcal{F}$, for each place $p \in D_F \cup J_F \cup F$, the *basic path expression* $\overline{\mathtt{expr}}_F(p)$ for p is defined by algorithm $\textsc{Build.Basic}$ in figure 2. The procedure partitions the set $\Pi_F(p)$ into sets of paths which enter place p from the same transition. Thus, the path set is partitioned into $k = |\ {}^{(t)}p\ |$ sets. Each set of paths is then used to create a product expression $\overline{\mathtt{minexpr}_i}$. Each product term represents a marked graph structure leading to the place p, as in [8]. These product expressions are then summed together to give $\overline{\mathtt{expr}}_F(p)$. Note from the procedure that each path in the $\overline{\mathtt{expr}}_F(p)$ is tagged as unexpanded.

Figure 2: Procedure $\textsc{Build.Basic}$ for constructing the basic expression for a place p

---

```
Procedure: BUILD.BASIC
For each forbidden condition $F$ and each place $p \in D_F \cup J_F \cup F$ {
    Define $k = |\ {}^{(t)}p\ |$ and index ${}^{(t)}p$ as ${}^{(t)}p = \{t_1, t_2, \dots t_k\}$
    For $1 \le i \le k$ {
        Define $\overline{\mathtt{minexpr}_i}$ as the unexpanded product of all paths
            in $\Pi_F(p)$ which enter place $p$ via transition $t_i$
    }
    Define $\overline{\mathtt{expr}}_F(p) \equiv \overline{\mathtt{minexpr}_1} + \overline{\mathtt{minexpr}_2} + \cdots + \overline{\mathtt{minexpr}_k}$
}
```

Expression Expansion

Once the basic expressions $\overline{\mathtt{expr}}(p)$ have been determined for all $p \in D_F \cup J_F \cup F$, they can be combined into a new expression to represent the structure of the uncontrolled subgraph. This is done by recursively replacing unexpanded elementary path terms $\overline{\pi}$ by expressions of paths that lead directly to the transition t_π. Given a forbidden condition F and a path $\pi \in \Pi_F$, define the expansion expression for a path π as:

$$\mathtt{expand}_F(\pi) \;\equiv\; \begin{cases} \pi & \text{if } t_\pi \in T_c \text{ or } {}^{(p)}t_\pi \cap \pi \neq \emptyset \\ (\pi + {}^{t_\pi}(\overline{\mathtt{expr}}_F(p_1)\overline{\mathtt{expr}}_F(p_2)\cdots\overline{\mathtt{expr}}_F(p_k)) & \text{else} \end{cases}$$

where each place p_i, $1 \leq i \leq k$, are the state places in $D_F \cup J_F \cup F$ which are inputs to transition t_π. The expression $\overline{\mathtt{expr}}_F(p_i)$ is the basic expression defined by the procedure BUILD.BASIC in the preceding section. Given any expression $\mathtt{expr} \in$ EXPR containing an unexpanded term $\overline{\pi}$, a *path expansion* for $\overline{\pi}$ in $\mathtt{expr}$ under F consists of replacing term $\overline{\pi}$ with expression $\mathtt{expand}_F(\pi)$. The expression $\mathtt{expr}_F(p)$ for a place p in a forbidden condition F is created from the unexpanded expression $\overline{\mathtt{expr}}_F(p)$ by recursively expanding paths until no more unexpanded paths exist.

Finally, the expression for a forbidden condition F can be defined as the product of the expressions for the places in F. For $F = \{p_1, p_2, \ldots p_n\}$, define

$$\mathtt{expr}(F) \equiv {}^{t_F}(\mathtt{expr}_F(p_1)\mathtt{expr}_F(p_2)\cdots\mathtt{expr}_F(p_n))$$

where t_F is an "imaginary" transition used in the path algebra. This transition is used solely to ensure proper manipulation and evaluation of the algebraic expression; it is not used to modify the original dynamics of the system. The places in the forbidden condition are the inputs to this transition, ${}^{(p)}t_F = F$. Without the use of the transition t_F in the expression $\mathtt{expr}(F)$, an assumption would be required that no place $p \in F$ lies within a path $\pi \in \Pi(\mathtt{expr}_F(p'))$ for some other $p' \in F$. This assumption, which we thus avoid, is referred to as the precedence path condition in earlier work [8, 9].

The expression characterizing the set $\mathcal{F} \equiv \{F_1, F_2, \ldots F_k\}$ of forbidden conditions is the sum of the expressions for the forbidden conditions in the set:

$$\mathtt{expr}(\mathcal{F}) \equiv \mathtt{expr}(F_1) + \mathtt{expr}(F_2) + \cdots + \mathtt{expr}(F_k)$$

Example: For the net and forbidden condition in figure 1, we have the following expressions:

$$\begin{aligned} \mathtt{expr}(p_4) &= (\pi_5 + {}^{t_3}(\pi_1 + \pi_2 \cdot (\pi_4 + {}^{t_5}\pi_3))) \\ \mathtt{expr}(p_5) &= (\pi_6 + {}^{t_5}\pi_3)(\pi_7 + {}^{t_6}\pi_3)(\pi_8)(\pi_9 + {}^{t_3}(\pi_1 + \pi_2 \cdot (\pi_4 + {}^{t_5}\pi_3))) \\ \mathtt{expr}(F) &= \mathtt{expr}(\mathcal{F}) = {}^{t_F}(\mathtt{expr}(p_4) \cdot \mathtt{expr}(p_5)) \end{aligned}$$

Token Threshold Calculation

The token threshold calculation is the last off-line procedure in preparation for the on-line control determination. In this procedure, path expressions are examined to determine the minimum number of tokens necessary in various paths for a forbidden marking to be uncontrollably reachable. In the subsequent on-line control determination, a control is determined to prevent the paths from achieving this minimum number of tokens.

The first step in token threshold calculation is to manipulate each expression $\mathtt{expr}(F)$ for each $F \in \mathcal{F}$ into a *sum-of-products normal form.* An expression $\mathtt{prod} \in$ EXPR is in *product form* if it does not contain any "+" operations, and is in *product normal form* if the product is in the form $e(\pi_1)e(\pi_2)\cdots e(\pi_n)$, where $\Pi(\mathtt{prod}) \equiv \{\pi_1, \pi_2, \ldots \pi_n\}$, and each $e(\pi_i)$ is of the form ${}^{\sigma_{i,1}}\pi_i\,{}^{\sigma_{i,2}}\pi_i \cdots {}^{\sigma_{i,k}}\pi_i$. A sum-of-products normal form expression is then of the form $\mathtt{prod}_1 + \mathtt{prod}_2 + \cdots + \mathtt{prod}_q$, where each $\mathtt{prod}_i$ is a product normal form expression. The manipulation of $\mathtt{expr}(F)$ into a sum-of-products normal form is accomplished by applying the commutative, associative, and distributive properties defined for the path algebra in [16].

For a product normal form expression $\mathtt{prod}$, define $\Sigma(\mathtt{prod}, \pi) \equiv \{\sigma \mid {}^{\sigma}\pi \text{ is a term in } \mathtt{prod}\}$ as the set of transition strings operating on π in $\mathtt{prod}$. A *conflict set* for path π and product $\mathtt{prod}$ is defined as any $\Sigma \subseteq \Sigma(\mathtt{prod}, \pi)$ such that for any $\sigma_1, \sigma_2 \in \Sigma$ where neither string is a suffix of the other, $\sigma_1 = \sigma_1' t_1 \sigma$ and $\sigma_2 = \sigma_2' t_2 \sigma$ with $t_1 \neq t_2$ and ${}^{(p)}t_1 \cap {}^{(p)}t_2 \neq \emptyset$. In words, σ_1 and σ_2 have the same suffix σ, but the transitions immediately preceding the common suffix in the strings are both the output transition of some state place. The *token threshold* $Q(\mathtt{prod}, \pi)$ for product $\mathtt{prod}$ and path $\pi \in \Pi(\mathtt{prod})$ is defined as the maximum cardinality $\mid \Sigma \mid$ over all conflict sets $\Sigma \subseteq \Sigma(\mathtt{prod}, \pi)$, where $Q(\mathtt{prod}, \pi) \equiv 1$ if $\pi \in \Pi(\mathtt{prod})$ but there are no conflict sets, and $Q(\mathtt{prod}, \pi) \equiv 0$ if $\pi \notin \Pi(\mathtt{prod})$.

The token threshold $Q(\mathtt{prod}, \pi)$ can be interpreted as follows. Each $\sigma \in \Sigma(\mathtt{prod}, \pi)$ indicates a sequence of transitions that a token in path π would follow to reach a forbidden marking. Given two sequences $\sigma_1, \sigma_2 \in \Sigma(\mathtt{prod}, \pi)$, if σ_1 and σ_2 are not in conflict, then a single token in π will branch at a transition into two or more tokens. These tokens can then concurrently follow the transition sequences σ_1 and σ_2 and can eventually lead to part of a forbidden marking. Transition strings which are not in conflict thus represent a marked graph structure, as considered in [8], and $Q({}^{\sigma_1}\pi\,{}^{\sigma_2}\pi) = 1$. If σ_1 and σ_2 are in conflict, then the point at which the sequences diverge (i.e. the transitions preceding the common suffix) represents an automata type structure. A single token in path π can only follow the sequence σ_1 or σ_2, but not both. In this case, $Q({}^{\sigma_1}\pi\,{}^{\sigma_2}\pi) = 2$, indicating that there must be at least two tokens in path π in order to reach the forbidden condition via transition strings σ_1 and σ_2. (Note: the introduction of the transition t_F in the previous section handles the situation when a forbidden place p is within a path leading toward a different forbidden place p' in F. The transition imposes a conflict between the transition t_F and the transition in $p^{(t)}$ leading toward p', and thus the threshold $Q(\cdot)$ will represent the number of tokens necessary to mark both p and p'.)

Example: For the net shown in figure 1 and the forbidden condition $F = \{p_4, p_5\}$, when $\mathtt{expr}(F)$ is converted into sum-of-products normal form, one product term is

$$\mathtt{prod} = ({}^{t_Ft_3}\pi_2)({}^{t_Ft_3t_5}\pi_3\,{}^{t_Ft_5}\pi_3\,{}^{t_Ft_6}\pi_3)({}^{t_F}\pi_8)$$

The transition strings t_Ft_5 and t_Ft_6 for π_3 are in conflict because both t_6 and t_5 are outputs of place p_8. However, $t_Ft_3t_5$ is not in conflict with t_Ft_5. The conflict sets are then $\{(t_Ft_3t_5), (t_Ft_6)\}$ and $\{(t_Ft_5), (t_Ft_6)\}$. Since both conflict sets have cardinality 2, then $Q(\mathtt{prod}, \pi_3) = 2$. For paths π_2 and π_8, there are no conflict sets, so $Q(\mathtt{prod}, \pi_2) = 1$ and $Q(\mathtt{prod}, \pi_8) = 1$.

4 On-line expression evaluation

The on-line operations consist of evaluating the path expressions for the current marking and determining the appropriate control. Two binary valuation functions are considered and are defined below. These valuations determine whether a place or forbidden condition is uncontrollably reachable or if a control is necessary.

Both valuation functions presented below are defined in terms of token thresholds of product form expressions. As indicated in the previous section, any expression can be converted into a sum-of-products form. The valuation functions are defined as follows:

Definition 2 : Given a marking $m \in \mathcal{M}$ for the net $\mathcal{G}$, the *control valuation* for m is a binary valuation $\Lambda_m : \textsc{Expr} \longrightarrow \{0, 1\}$, for any product expression $\mathtt{prod}$ in $\textsc{Expr}$,

$$\Lambda_m(\mathtt{prod})) \equiv 1 \text{ iff } m(\pi) \geq Q(\mathtt{prod}, \pi)) \text{ for all } \pi \in \Pi(\mathtt{prod}).$$

Definition 3 : Given a marking $m \in \mathcal{M}$ for the net $\mathcal{G}$, the *boundary valuation* for m is a binary valuation $\Delta_m : \textsc{Expr} \longrightarrow \{0, 1\}$ such that for any product expression $\mathtt{prod}$ in $\textsc{Expr}$, $\Delta_m(\mathtt{prod}) \equiv 1$ iff for all $\pi \in \Pi(\mathtt{prod})$:

1. $m(\pi) \geq (Q(\mathtt{prod}, \pi) - 1)$ if $t_\pi \in T_c$ and t_π is state enabled under m;

2. $m(\pi) \geq Q(\texttt{prod}, \pi)$ otherwise.

Note that the definition of Δ above is significantly different from the definition used in [8, 9]. In particular, in the above definition for any $\texttt{expr} \in \textsc{Expr}$ and any $m \in \mathcal{M}$, $\Lambda_m(\texttt{expr}) = 1$ implies $\Delta_m(\texttt{expr}) = 1$.

Both valuation definitions can be extended to sum-of-products form by defining the "+" operator to be disjunction. Thus, $\Lambda(\texttt{expr}_1 + \texttt{expr}_2) = 1$ if and only if $\Lambda(\texttt{expr}_1) = 1$ or $\Lambda(\texttt{expr}_2) = 1$. Similarly, $\Delta(\texttt{expr}_1 + \texttt{expr}_2) = 1$ if and only if $\Delta(\texttt{expr}_1) = 1$ or $\Delta(\texttt{expr}_2) = 1$. Consequently, the valuations can be applied to any expression in $\textsc{Expr}$ by converting it into sum-of-products form and applying the above definition of the "+" operator.

For each $F \in \mathcal{F}$, the set of controls for F under m is:

$$\mathcal{C}_m(\texttt{prod}) \equiv \begin{cases} \emptyset & \text{if } \Lambda_m(\texttt{prod}) = 1 \text{ or } \Delta_m(\texttt{prod}) = 0 \\ \{c \in {}^{(c)}t_\pi \mid \pi \in \Pi(\texttt{prod}), \Lambda_m(\texttt{e}(\pi)) = 0, \Delta_m(\texttt{e}(\pi)) = 1\} & \text{else} \end{cases}$$

This allows us to define the *nondeterministic control policy* U_F^* as follows: For any marking m, $u \in U_{\mathcal{F}}^*(m)$ if for each $F \in \mathcal{F}$ and each product term $\texttt{prod}$ in the sum-of-products form of $\texttt{expr}(\mathcal{F})$ with $\mathcal{C}_m(\texttt{prod}) \neq \emptyset$, $u(c) = 0$ for at least one $c \in \mathcal{C}_m(\texttt{prod})$. The controller then enforces some $u \in U_{\mathcal{F}}^*(m)$ for the given marking m. In the following section, we state analytical results showing that this is the maximal set of controls suitable for preventing the reachability of forbidden markings under some basic assumptions.

5 Analytical Results

The previous section outlined a method for feedback control for avoiding forbidden states in controlled Petri nets. In this section, we summarize several results regarding the control policy and the algebraic expressions used. The results depend upon the following conditions on the net and the forbidden conditions.

Definition 4 : A Petri net $\mathcal{G}$ with forbidden condition set $\mathcal{F}$ is said to satisfy the *net conditions* if the following conditions are satisfied:

1. Net structure conditions:
   (a) For any trap $P \subseteq \mathcal{P}$, there exists at least one place $p \in P$ such that $p^{(t)} \cap T_c \neq \emptyset$;
   (b) $\mathcal{G}$ does not contain any self-loops;
2. Path interaction conditions:
   (a) For any $p \in D_F$ and $\pi \in \Pi(p)$, there exists at most one $p' \in {}^{(p)}(p^{(t)}) \cap (D_F \cup J_F \cup F)$ for which $t_{\pi'} = t_\pi$ for some $\pi' \in \Pi(p')$.
   (b) For any place $p \in (D_F \cup J_F \cup F)$ and paths $\pi_1, \pi_2 \in \Pi(p)$, if $t_{\pi_1} = t_{\pi_2}$, then both paths enter place p via the same transition.
3. Path input conditions:
   (a) For any $\pi_1, \pi_2 \in \Pi(F)$ with $t_{\pi_1} \neq t_{\pi_2}$, $({}^{(p)}t_{\pi_1} \cap {}^{(p)}t_{\pi_2}) = \emptyset$.
   (b) For any path $\pi \in \Pi(F)$ with $t_\pi \in T_c$, ${}^{(p)}t_\pi \cap P_F = \emptyset$.

Each of the conditions listed above are rather mild and either do not significantly restrict the class of nets considered or can be easily circumvented by a small modification of the net. The net structure conditions above are straightforward (see [16] or [17] for definitions of self-loops and traps in Petri nets). The path interaction conditions both pertain to paths which begin from the same transition and end either at the same place or at places which share an output transition. Both conditions are important for insuring that the valuations on the algebraic expressions will correctly characterize the reachability of forbidden markings. The path input conditions above are similar to the precedence path input conditions (PPIC) in earlier papers [8, 9]. These are necessary for properly determining when a control must be disabled.

The valuations on the path algebra expressions can now be interpreted using the following theorems from [16].

Theorem 1 : [16] Given a CtlPN $\mathcal{G}$ and a forbidden condition F satisfying Net Conditions 1 and 2, for any marking $m \in \mathcal{M}$, $m \in W(M_F)$ if and only if $\Lambda_m(\texttt{expr}(F)) = 1$.

Theorem 2 : [16] Given a CtlPN $\mathcal{G}$ and a forbidden condition F satisfying all of the Net Conditions, for any marking $m \in \mathcal{M}$, $m \in \delta(W(M_F))$ if and only if $\Delta_m(\texttt{expr}(F)) = 1$ and $\Lambda_m(\texttt{expr}(F)) = 0$.

Theorem 1 shows that the expression $\texttt{expr}(F)$ can be used to characterize the weakly forbidden marking set for a given forbidden condition F. Theorem 2 is used to characterize the set of markings $\delta(W(M_F))$ for which a control must be enforced to prevent the system from entering a marking which could uncontrollably lead to a forbidden marking.

Finally, we consider the control law $U^*_{\mathcal{F}}$ defined in the previous section. The following theorem follows directly from the results in [16]:

Theorem 3 : Given a CtlPN $\mathcal{G}$, a set of forbidden conditions $\mathcal{F}$ satisfying the Net Conditions, and a marking $m \in \mathcal{M} - W(M_{\mathcal{F}})$, the following statements are true:

1. For any $u \in U^*_{\mathcal{F}}(m)$, all immediately reachable markings $R_1(m, u)$ under u are admissible markings, i.e. $R_1(m, u) \subseteq (\mathcal{M} - W(M_{\mathcal{F}}))$.
2. A control $u \in \mathcal{U}$ is a maximally permissive control if and only if (1.) $u \in U^*_{\mathcal{F}}(m)$ and (2.) for each $c \in \mathcal{C}$ for which $u(c) = 0$, there exists some $F \in \mathcal{F}$ and some $\texttt{prod}$ in the sum-of-products form of $\texttt{expr}(F)$ such that $\Delta_m(\texttt{prod}) = 1$ and $u(c') = 1$ for all $c' \in \mathcal{C}_m(\texttt{prod}) - \{c\}$.

6 Discussion

This paper described a method of feedback control for forbidden state avoidance in controlled Petri nets. The method uses path algebra expressions to represent the structure of the Petri net and the conditions for reachability of forbidden states. The expressions are evaluated on-line for a current marking to determine the maximal set of admissible controls to ensure that no forbidden marking will be reached.

References

[1] P. J. Ramadge and W. M. Wonham. Supervisory control of a class of discrete-event processes. *SIAM J. on Control an Optimization*, 25, Jan 1987.

[2] P. J. Ramadge and W. M. Wonham. Modular feedback logic for discrete-event systems. *SIAM J. of Control and Optimization*, 25, Sept 1987.

[3] Michael Heymann. Concurrency and discrete event control. *IEEE Control Systems Magazine*, 10(4):103–112, June 1990.

[4] K. Inan and P. Varaiya. Finitely recursive process models for discrete event systems. *IEEE Transactions on Automatic Control*, 33(7), July 1988.

[5] A. Ichikawa and K. Hiraishi. Analysis and control of discrete event systems represented by Petri nets. *Discrete Event Systems: Models and Applications*, IIASA Conference, Sopron, Hungary, August 3-7, 1987, 1988. Springer-Verlag, New York.

[6] B. H. Krogh. Controlled Petri nets and maximally permissive feedback logic. *Proceedings of 25th Annual Allerton Conference*, Sept. 1987. University of Illinois, Urbana.

[7] Y. Li and W.M. Wonham. Control of vector discrete-event systems i – the base model. *IEEE Transactions on Automatic Control*, 38(8):1214–1227, August 1993.

[8] L. E. Holloway and B. H. Krogh. Synthesis of feedback control logic for a class of controlled Petri nets. *IEEE Transactions on Automatic Control*, 35(5):514–523, May 1990. Also appears in *Discrete Event Dynamic Systems: Analyzing Complexity and Performance in the Modern World*, edited by Y.C. Ho, IEEE Press, New York, 1992.

[9] Bruce H. Krogh and Lawrence E. Holloway. Synthesis of feedback control logic for discrete manufacturing systems. *Automatica*, July 1991.

[10] L. E. Holloway and B. H. Krogh. On closed-loop liveness of discrete event systems under maximally permissive control. *IEEE Transactions on Automatic Control*, 37(5), May 1992.

[11] L. Ben-Naoum, V. Van Breusegem, and R.K. Boel. Synthesis via decomposition of feedback control logic for a class of controlled petri nets. In *Proceedings of the 1993 Conference on Decision and Control*, San Antonio, Texas, December 1993. submitted.

[12] Ratnesh Kumar and Lawrence Holloway. Supervisory control of petri net languages. In *Proceedings of the 31st IEEE Conference on Decision and Control*, pages 1190–1195, Tucson, Arizona, December 1992.

[13] L.E. Holloway and F. Hossain. Feedback control for sequencing specifications in Controlled Petri Nets. In *Third International Conference on Computer Integrated Manufacturing*, pages 242–250, Troy, New York, May 1992.

[14] Archana Sathaye and Bruce H. Krogh. Logical analysis and control of time petri nets. In *Proceedings of the 1992 IEEE Conference on Decision and Control*, 1992.

[15] Y. Brave and B.H. Krogh. Maximally permissive policies for controlled marked graphs. In *Proceedings of 12th IFAC World Congress*, Sydney, July 1993.

[16] L.E. Holloway and X. Guan. A generalization of state avoidance policies for controlled petri nets. In *Proceedings of 32nd IEEE Conference on Decision and Control*, San Antonio, Texas, December 1993.

[17] Tadao Murata. Petri nets: Properties, analysis and applications. *Proceedings of the IEEE*, 77(4):541–580, April 1989.

Non-blocking Control of Nondeterministic Systems *

Ratnesh Kumar
Department of Electrical Engineering
University of Kentucky
Lexington, KY 40506-0046
Email: kumar@engr.uky.edu

Mark A. Shayman
Department of Electrical Engineering and
Institute for Systems Research
University of Maryland
College Park, MD 20742
Email: shayman@eng.umd.edu

Invited Paper

Abstract

In a previous paper [16], we showed that supervisory control of nondeterministic discrete event systems, in the presence of driven events, can be achieved using prioritized synchronous composition as a mechanism of control, and trajectory models as a modeling formalism. The specifications considered in [16] were given by *prefix-closed* languages. In this paper, we extend the theory of trajectory models and prioritized synchronous composition to include markings so that non-closed specifications and issues such as blocking can be addressed. It is shown that the usual notion of non-blocking, called *language model non-blocking*, is inadequate in the setting of nondeterministic systems, and a stronger notion, called *trajectory model non-blocking*, is introduced. Necessary and sufficient conditions for the existence of non-marking and language model non-blocking as well as trajectory model non-blocking supervisors is obtained for nondeterministic systems in the presence of driven events. We also show that our approach is also suitable for modular supervisory control.

1 Introduction

The majority of the research effort in the area of supervisory control of discrete event systems has been focused on *deterministic* systems, and relatively little progress has been made towards *nondeterministic* systems–systems in which knowledge of the current state and next event is insufficient to uniquely determine the next state. Nondeterminism in the context of supervisory control arises due to unmodeled system dynamics such as suppression of "internal" events, suppression of "ticks of the clock" to obtain the untimed model from a timed model [2], partial observation of the system behavior due to lack of enough sensors or presence of faulty sensors, etc.

In the Ramadge-Wonham approach to supervisory control, every event is generated by the plant and synchronously executed by the supervisor [14] which acts passively by disabling certain controllable events possible in the open-loop plant. The disablement action is accomplished by a control-input map which specifies a set of disabled events based on the current state of the supervisor. Alternatively, in the work of Kumar-Garg-Marcus [10], the disablement action is accomplished by removing certain transitions from the structure of the supervisor while continuing to require that the plant and supervisor be connected by strict synchronous composition (SSC). In the work of Golaszewski-Ramadge [3] and, in the real-time setting, the work of Brandin-Wonham [2], the supervisor is able to initiate certain so-called *forcible* events that the plant synchronously executes. In the work of Balemi and coworkers [1], events can originate in the supervisor (so-called *command events*) or in the plant (so-called *response events*). The assumption is made that the plant and supervisor are *mutually receptive*, meaning that neither the plant nor the supervisor can refuse to execute an event initiated by the other.

Common to all of the above approaches is the assumption that there are never events which may occur in the supervisor without the participation of the plant. However, this assumption may be unreasonably restrictive for nondeterministic systems. When the plant is nondeterministic, there is generally no way to know a priori whether a command issued by the supervisor can be executed by the plant in its current state.

---

*This research was supported in part by the Center for Robotics and Manufacturing, University of Kentucky, in part by by the National Science Foundation under the Engineering Research Centers Program Grant CDR-8803012, the Minta Martin Fund for Aeronautical Research, and the General Research Board at the University of Maryland.

For example, it may be impossible to know that a device is in a faulted state until after it fails to respond to a command from the controller.

Heymann has introduced an interconnection operator called *prioritized synchronous composition* (PSC) [4], which relaxes the synchronization requirements between the plant and supervisor. Each process in a PSC-interconnection is assigned a *priority set* of events. For an event to be enabled in the interconnected system, it must be enabled in all processes whose priority sets contain that event. Also, when an enabled event occurs, it occurs in each subsystem in which the event is enabled. In the context of supervisory control, the priority set of the plant contains the controllable and uncontrollable events, while the priority set of the supervisor contains the controllable and driven events. Thus, controllable events require the participation of both plant and supervisor; uncontrollable events require the participation of the plant and will occur synchronously in the supervisor whenever possible; driven events require the participation of the supervisor and will occur synchronously in the plant whenever possible.

It is important to distinguish between PSC and other types of parallel composition in the literature. For example, Hoare [6] defines a concurrent composition operator in which each process has its own alphabet and the processes synchronize on the events in the intersection of their alphabets. This is generalized to trace-dependent alphabets, called event-control sets, by Inan-Varaiya [9]. The key difference between concurrent composition and PSC is that in PSC, although a process cannot block events which are outside its priority set, it may be able to execute these events–and, whenever possible, will execute these events synchronously when they occur in the other process[1].

Language models identify processes that have the same set of traces. The failures model of Hoare [6] identifies processes that have the same set of so-called *failures*. Failure equivalence refines language equivalence. Heymann showed that failure equivalence is too coarse to support the PSC operator [4]. In other words, there exist plants $\mathcal{P}_1$, $\mathcal{P}_2$ with the same failures model (and hence with the same language model) such that their PSC's $\mathcal{P}_1\,{}_A\|_B\,\mathcal{S}$, $\mathcal{P}_2\,{}_A\|_B\,\mathcal{S}$ with a common supervisor have different language models. Thus, neither the language model nor even the failures model retains enough information about a process to do control design using the operation of PSC.

This has led Heymann to introduce the *trajectory model*, a refinement of the failures model [4, 5]. The trajectory model is similar to the *failure-trace model* (also called the refusal-testing model) in concurrency theory [13], but differs from this model in its treatment of hidden transitions. The trajectory model treats hidden transitions in a way that is consistent with the failures model. In previous papers [16, 18, 17], we proved that if $\mathcal{P}_1$, $\mathcal{P}_2$ and $\mathcal{S}_1$, $\mathcal{S}_2$ are nondeterministic state machines (with ϵ-transitions), and if the pairs $\mathcal{P}_1$, $\mathcal{P}_2$ and $\mathcal{S}_1$, $\mathcal{S}_2$ each have the same trajectory models, then $\mathcal{P}_1\,{}_A\|_B\,\mathcal{S}_1$ and $\mathcal{P}_2\,{}_A\|_B\,\mathcal{S}_2$ have the same trajectory model–and hence the same language model as well. Thus, the trajectory model retains sufficient process detail to permit PSC-based controller design.

In [16], we showed that supervisory control of nondeterministic discrete event systems, in the presence of driven events, can be achieved using prioritized synchronous composition as a mechanism of control, and trajectory models as a modeling formalism. The specifications considered in [16] were given by *prefix-closed* languages. In this paper, we extend our previous work on supervisory control of nondeterministic systems using trajectory models and prioritized synchronous composition to include the notion of markings, by introducing the notion of recognized and generated trajectory sets, so that non-closed specifications and issues such as blocking can be addressed.

The usual notion of non-blocking, referred to as *language model non-blocking* in this paper, requires that each trace belonging to the generated language of a controlled system be extendable to a trace belonging to the recognized language. This property adequately captures the notion of non-blocking in a deterministic setting. However, the execution of a trace belonging to the generated behavior of a controlled nondeterministic system may lead to more than one state. Language model non-blocking only requires that every such trace be extendable to a trace in the recognized behavior from *at least one* such state–as opposed to *all* such states. Thus, a language model non-blocking nondeterministic system can deadlock, as illustrated by the example in the next section. Consequently, there is a need for a stronger type of non-blocking for nondeterministic systems. This leads us to introduce the property of *trajectory model non-blocking*, which requires that each refusal-trace belonging to the generated trajectory set of a controlled nondeterministic system be extendable to a refusal-trace belonging to the recognized trajectory set.

Another desirable property of a supervisor is that it should be *non-marking*, i.e., given a trace (respectively, a refusal-trace) of the controlled system, it should belong to the recognized language (respectively, the recognized trajectory set) of the controlled system if and only if a marked state of the uncontrolled system is reached due to its execution (regardless of the type of state reached in the supervisor). We first obtain a necessary and sufficient condition for the existence of a non-marking and language model non-blocking supervisor for a given nondeterministic system in the presence of driven events. This result is then used to obtain a necessary and sufficient condition for the existence of a non-marking and trajectory model non-blocking supervisor in that setting. We also demonstrate that our approach is suitable for modular supervisory control.

---

[1] If applied to so-called *improper* processes, the parallel operator defined by Inan [8] can be viewed as a generalized form of PSC, but only in the deterministic setting. However, when supervisory control is considered in this reference, the assumption is made that the plant is proper and has a constant event control set. This assumption excludes driven events.

Due to lack of space none of the proofs are included; they can be found in [11].

2 Notation and Preliminaries

Given a finite event set Σ, Σ^* is used to denote the collection of all *traces*, i.e., finite sequences of events, including the zero length sequence, denoted by ϵ. A subset of Σ^* is called a language. Symbols H, K, etc. are used to denote languages. The set $2^\Sigma \times (\Sigma \times 2^\Sigma)^*$ is used to denote the collection of all *refusal-traces*, i.e., finite sequences of alternating *refusals* and events [5, 16] of the type:

$$\Sigma_0(\sigma_1, \Sigma_1)\dots(\sigma_n, \Sigma_n),$$

where $n \in \mathcal{N}$. The sequence $\sigma_1 \dots \sigma_n \in \Sigma^*$ is the trace, and for each $i \le n$, $\Sigma_i \subseteq \Sigma$ is the set of events refused (if offered) at the indicated point. Symbols P, Q, R, S, etc. are used to denote sets of refusal-traces. Refusal-traces are also referred to as *trajectories*.

Given $s \in \Sigma^*$, we use $|s|$ to denote the length of s, and for each $k \le |s|$, $\sigma_k(s) \in \Sigma$ is used to denote the kth event in s. If $t \in \Sigma^*$ is another trace such that $|t| \le |s|$ and for each $k \le |t|$, $\sigma_k(t) = \sigma_k(s)$, then t is said to be a prefix of s, denoted $t \le s$. For each $k \le |s|$, s^k denotes the prefix of length k of s. The prefix-closure of $s \in \Sigma^*$, denoted $pr(s) \subseteq \Sigma^*$, is defined as $pr(s) := \{t \in \Sigma^* \mid t \le s\}$. The prefix-closure map can be defined for a set of traces in a natural way. Given $e \in 2^\Sigma \times (\Sigma \times 2^\Sigma)^*$, we use $|e|$ to denote the length of e, and for each $k \le |e|$, $\Sigma_k(e) \subseteq \Sigma$ is used to denote the kth refusal in e and $\sigma_k(e) \in \Sigma$ is used to denote the kth event in e, i.e.,

$$e = \Sigma_0(e)(\sigma_1(e), \Sigma_1(e))\dots(\sigma_k(e), \Sigma_k(e))\dots(\sigma_{|e|}(e), \Sigma_{|e|}(e)).$$

If $f \in 2^\Sigma \times (\Sigma \times 2^\Sigma)^*$ is another refusal-trace such that $|f| \le |e|$ and for each $k \le |f|$, $\Sigma_k(f) = \Sigma_k(e)$ and $\sigma_k(f) = \sigma_k(e)$, then f is said to be a prefix of e, denoted $f \le e$. For each $k \le |e|$, e^k is used to denote the prefix of length k of e. If $f \in 2^\Sigma \times (\Sigma \times 2^\Sigma)^*$ is such that $|f| = |e|$ and for each $k \le |f|$, $\Sigma_k(f) \subseteq \Sigma_k(e)$ and $\sigma_k(f) = \sigma_k(e)$, then f is said to be dominated by e, denoted $f \sqsubseteq e$. The prefix-closure of $e \in 2^\Sigma \times (\Sigma \times 2^\Sigma)^*$, denoted $pr(e) \subseteq 2^\Sigma \times (\Sigma \times 2^\Sigma)^*$, is defined as $pr(e) := \{f \in 2^\Sigma \times (\Sigma \times 2^\Sigma)^* \mid f \le e\}$, and the dominance-closure of e, denoted $dom(e) \subseteq 2^\Sigma \times (\Sigma \times 2^\Sigma)^*$, is defined as $dom(e) := \{f \in 2^\Sigma \times (\Sigma \times 2^\Sigma)^* \mid f \sqsubseteq e\}$. The prefix-closure and dominance-closure maps can be defined for a set of refusal-traces in a natural way. Given a refusal-trace $e \in 2^\Sigma \times (\Sigma \times 2^\Sigma)^*$, the trace of e, denoted $tr(e) \in \Sigma^*$, is defined as $tr(e) := \sigma_1(e)\dots\sigma_{|e|}(e)$. The trace map can be extended to a set of refusal-traces in a natural way. Given a set of refusal-traces $P \subseteq 2^\Sigma \times (\Sigma \times 2^\Sigma)^*$, we use $L(P) := tr(P)$ to denote its set of traces.

Symbols $\mathcal{P}, \mathcal{Q}, \mathcal{R}$, etc. are used to denote NSM's (with ϵ-moves). Let the 5-tuple

$$\mathcal{P} := (X_\mathcal{P}, \Sigma, \delta_\mathcal{P}, x^0_\mathcal{P}, X^m_\mathcal{P})$$

represent a discrete event system modeled as an NSM, where $X_\mathcal{P}$ is the state set, Σ is the finite event set, $\delta_\mathcal{P} : X_\mathcal{P} \times (\Sigma \cup \{\epsilon\}) \longrightarrow 2^{X_\mathcal{P}}$ denotes the nondeterministic transition function[2], $x^0_\mathcal{P} \in X_\mathcal{P}$ is the initial state, and $X^m_\mathcal{P} \subseteq X_\mathcal{P}$ is the set of accepting or marked states. A triple $(x_1, \sigma, x_2) \in X_\mathcal{P} \times (\Sigma \cup \{\epsilon\}) \times X_\mathcal{P}$ is said to be a transition if $x_2 \in \delta_\mathcal{P}(x_1, \sigma)$. A transition (x_1, ϵ, x_2) is referred to as a *silent* or *hidden* transition. We assume that the plant cannot undergo an unbounded number of silent transitions, i.e., $\mathcal{P}$ does not contain any cycle of silent transitions. The *ϵ-closure* of $x \in X_\mathcal{P}$, denoted $\epsilon^*_\mathcal{P}(x) \subseteq X_\mathcal{P}$, is defined recursively as

$$x \in \epsilon^*_\mathcal{P}(x), \text{ and } x' \in \epsilon^*_\mathcal{P}(x) \Rightarrow \delta_\mathcal{P}(x', \epsilon) \subseteq \epsilon^*(x),$$

and the set of *refusal events* at $x \in X_\mathcal{P}$, denoted $\Re_\mathcal{P}(x) \subseteq \Sigma$, is defined as

$$\Re_\mathcal{P}(x) := \{\sigma \in \Sigma \mid \delta_\mathcal{P}(x', \sigma) = \emptyset, \forall x' \in \epsilon^*_\mathcal{P}(x)\}.$$

In other words, given $x \in X_\mathcal{P}$, $\epsilon^*_\mathcal{P}(x)$ is the set of states that can be reached from x on zero or more ϵ-moves, and $\Re_\mathcal{P}(x)$ is the set of events that are undefined at each state in the ϵ-closure of x.

---

[2] ϵ represents both an *internal* or *unobservable* event and an *internal* or *nondeterministic* choice [6, 12].

The transition function $\delta_{\mathcal{P}} : X \times (\Sigma \cup \{\epsilon\}) \rightarrow 2^{X_{\mathcal{P}}}$ is extended to the set of *traces* as $\delta^*_{\mathcal{P}} : X \times \Sigma^* \rightarrow 2^{X_{\mathcal{P}}}$, which is defined inductively as:

$$\forall x \in X_{\mathcal{P}} : \begin{cases} \delta^*_{\mathcal{P}}(x, \epsilon) := \epsilon^*_{\mathcal{P}}(x), \\ \forall s \in \Sigma^*, \sigma \in \Sigma : \delta^*_{\mathcal{P}}(x, s\sigma) := \epsilon^*_{\mathcal{P}}(\delta_{\mathcal{P}}(\delta^*_{\mathcal{P}}(x, s), \sigma)), \end{cases}$$

where in the last inequality, the transition function $\delta_{\mathcal{P}} : X \times (\Sigma \cup \{\epsilon\}) \rightarrow 2^{X_{\mathcal{P}}}$ has been extended to $\delta_{\mathcal{P}} : 2^{X_{\mathcal{P}}} \times (\Sigma \cup \{\epsilon\}) \rightarrow 2^{X_{\mathcal{P}}}$ in a natural way. The transition function is also extended to the set of *refusal-traces* as $\delta^T_{\mathcal{P}} : X \times (2^{\Sigma} \times (\Sigma \times 2^{\Sigma})^*) \rightarrow 2^{X_{\mathcal{P}}}$, which is defined inductively as:

$$\forall x \in X_{\mathcal{P}} : \begin{cases} \forall \Sigma' \subseteq \Sigma : \delta^T_{\mathcal{P}}(x, \Sigma') := \{x' \in \epsilon^*_{\mathcal{P}}(x) \mid \Sigma' \subseteq \Re_{\mathcal{P}}(x')\}, \\ \forall e \in 2^{\Sigma} \times (\Sigma \times 2^{\Sigma})^*, \sigma \in \Sigma, \Sigma' \subseteq \Sigma : \\ \delta^T_{\mathcal{P}}(x, e(\sigma, \Sigma')) := \{x' \in \epsilon^*_{\mathcal{P}}(\delta_{\mathcal{P}}(\delta^T_{\mathcal{P}}(x, e), \sigma)) \mid \Sigma' \subseteq \Re_{\mathcal{P}}(x')\}. \end{cases}$$

These maps are then used to obtain the language models and the trajectory models of $\mathcal{P}$ as follows:

$$L(\mathcal{P}) := \{s \in \Sigma^* \mid \delta^*_{\mathcal{P}}(x^0_{\mathcal{P}}, s) \neq \emptyset\}, \quad L^m(\mathcal{P}) := \{s \in L(\mathcal{P}) \mid \delta^*_{\mathcal{P}}(x^0_{\mathcal{P}}, s) \cap X^m_{\mathcal{P}} \neq \emptyset\},$$

$$T(\mathcal{P}) := \{e \in 2^{\Sigma} \times (\Sigma \times 2^{\Sigma})^* \mid \delta^T_{\mathcal{P}}(x^0_{\mathcal{P}}, e) \neq \emptyset\}, T^m(\mathcal{P}) := \{e \in T(\mathcal{P}) \mid \delta^T_{\mathcal{P}}(x^0_{\mathcal{P}}, e) \cap X^m_{\mathcal{P}} \neq \emptyset\}.$$

$L(\mathcal{P}), L^m(\mathcal{P}), T(\mathcal{P}), T^m(\mathcal{P})$ are called the *generated language*, *recognized language*, *generated trajectory set*, *recognized trajectory set*, respectively, of $\mathcal{P}$. It is easily seen that $L(T^m(\mathcal{P})) = L^m(\mathcal{P})$ and $L(T(P)) = L(\mathcal{P})$. The pairs $(L^m(\mathcal{P}), L(\mathcal{P}))$ and $(T^m(\mathcal{P}), T(\mathcal{P}))$ are called the language model and the trajectory model, respectively, of $\mathcal{P}$. Two language models $(K^m_1, K_1), (K^m_2, K_2)$ are said to be equal, written $(K^m_1, K_1) = (K^m_2, K_2)$, if $K^m_1 = K^m_2, K_1 = K_2$; equality of two trajectory models is defined analogously.

3 Trajectory Models and Prioritized Synchronization

It is clear that a language pair (K^m, K) with $K^m, K \subseteq \Sigma^*$ is a language model if and only if

$$K \neq \emptyset, \text{ and } K^m \subseteq K = pr(K).$$

We next obtain a necessary and sufficient condition for a given refusal-trace set pair to be a trajectory model. This requires the definition of saturated refusal-traces. Given a refusal-trace set $P \subseteq 2^{\Sigma} \times (\Sigma \times 2^{\Sigma})^*$, we define the *saturation map* on P by $sat_P : P \rightarrow 2^{\Sigma} \times (\Sigma \times 2^{\Sigma})^*$ where

$$\begin{aligned} sat_P(e) &:= \Sigma_0(\sigma_1(e), \Sigma_1) \ldots (\sigma_{|e|}, \Sigma_{|e|}), \\ \Sigma_k &:= \Sigma_k(e) \cup \{\sigma \in \Sigma \mid e^k(\sigma, \emptyset) \notin dom(pr(P))\}, \forall k \leq |e|. \end{aligned}$$

The *saturated refusal-traces* of P, denoted $P_{sat} \subseteq P$, is defined to be the set of fixed points of $sat_P(\cdot)$. The following result characterizes those refusal-trace set pairs that are trajectory models of NSM's.

Theorem 1 [11] Given a pair of refusal-trace sets (P^m, P) with $P^m, P \subseteq 2^{\Sigma} \times (\Sigma \times 2^{\Sigma})^*$, it is a trajectory model if and only if

T1: $P \neq \emptyset$
T2: $P = pr(P)$
T3: $P = dom(P_{sat})$
T4: $\forall e \in P : \sigma_{k+1}(e) \notin \Sigma_k(e), \quad k \leq |e| - 1$
T5: $P^m = dom(P_{sat} \cap P^m)$

Note that given a trajectory model, the trace map can be used to obtain the associated language model. On the other hand, given a language model (K^m, K), it is possible to obtain a *deterministic* trajectory model having (K^m, K) as the language model.

Definition 1 A trajectory model (P^m, P) is said to be *deterministic* if there exists a deterministic state machine $\mathcal{P} := (X_{\mathcal{P}}, \Sigma, \delta_{\mathcal{P}}, x^0_{\mathcal{P}}, X^m_{\mathcal{P}})$ such that $T^m(\mathcal{P}) = P^m$ and $T(\mathcal{P}) = P$.

Given a language model (K^m, K), the trajectory map $trj_K : K \rightarrow 2^{\Sigma} \times (\Sigma \times 2^{\Sigma})^*$ can be used to obtain the associated deterministic trajectory model:

$$\begin{aligned} trj_K(s) &:= \Sigma_0(s)(\sigma_1(s), \Sigma_1(s)) \ldots (\sigma_{|s|}(s), \Sigma_{|s|}(s)) \in 2^{\Sigma} \times (\Sigma \times 2^{\Sigma})^*, \text{ where} \\ \Sigma_k(s) &:= \{\sigma \in \Sigma \mid s^k\sigma \notin K\}, \forall k \leq |s|. \end{aligned}$$

Let $det(K) := dom(trj_K(K))$ and $det^m(K^m, K) := dom(trj_K(K^m))$.

Proposition 1 [11] Given a language model (K^m, K), $(det^m(K^m, K), det(K))$ is the unique deterministic trajectory model with language model (K^m, K).

In [4, 5, 16, 18, 17], prioritized synchronous composition (PSC) of systems is used as the mechanism of control. In this setting, each system is assigned a priority set of events. When systems are interconnected via PSC, an event can occur in the composite system only if it can occur in each subsystem which has that event in its priority set. In this way, a subsystem can prevent the occurrence of certain events, thereby implementing a type of supervisory control. The formal definition of the PSC of two NSM's is as follows:

Definition 2 Let $\mathcal{P} := (X_{\mathcal{P}}, \Sigma, \delta_{\mathcal{P}}, x^0_{\mathcal{P}}, X^m_{\mathcal{P}})$, $\mathcal{Q} := (X_{\mathcal{Q}}, \Sigma, \delta_{\mathcal{Q}}, x^0_{\mathcal{Q}}, X^m_{\mathcal{Q}})$ be two NSM's having priority sets $A, B \subseteq \Sigma$ respectively. The PSC of $\mathcal{P}$ and $\mathcal{Q}$ is another NSM which is denoted by

$$\mathcal{P}\ {}_A\|_B\ \mathcal{Q} := \mathcal{R} := (X_{\mathcal{R}}, \Sigma, \delta_{\mathcal{R}}, x^0_{\mathcal{R}}, X^m_{\mathcal{R}}),$$

where $X_{\mathcal{R}} = X_{\mathcal{P}} \times X_{\mathcal{Q}}, x^0_{\mathcal{R}} = (x^0_{\mathcal{P}}, x^0_{\mathcal{Q}}), X^m_{\mathcal{R}} = X^m_{\mathcal{P}} \times X^m_{\mathcal{Q}}$, and the state transition function $\delta_{\mathcal{R}} : X_{\mathcal{R}} \times (\Sigma \cup \{\epsilon\}) \rightarrow 2^{X_{\mathcal{R}}}$ is defined as:

$\forall x_r = (x_p, x_q) \in X_{\mathcal{R}}$:

$$\begin{aligned}
\forall \sigma \in \Sigma : \delta_{\mathcal{R}}(x_r, \sigma) &:= \begin{cases} \delta_{\mathcal{P}}(x_p, \sigma) \times \delta_{\mathcal{Q}}(x_q, \sigma) & \text{if } \delta_{\mathcal{P}}(x_p, \sigma), \delta_{\mathcal{Q}}(x_q, \sigma) \neq \emptyset \\ \delta_{\mathcal{P}}(x_p, \sigma) \times \{x_q\} & \text{if } \delta_{\mathcal{P}}(x_p, \sigma) \neq \emptyset, \sigma \in \Re_{\mathcal{Q}}(x_q), \sigma \notin B \\ \{x_p\} \times \delta_{\mathcal{Q}}(x_q, \sigma) & \text{if } \delta_{\mathcal{Q}}(x_q, \sigma) \neq \emptyset, \sigma \in \Re_{\mathcal{P}}(x_p), \sigma \notin A \\ \emptyset & \text{otherwise,} \end{cases} \\
\delta_{\mathcal{R}}(x_r, \epsilon) &:= [\delta_{\mathcal{P}}(x_p, \epsilon) \cup \{x_p\}] \times [\delta_{\mathcal{Q}}(x_q, \epsilon) \cup \{x_q\}] - \{(x_p, x_q)\}.
\end{aligned}$$

This definition of the PSC of two NSM's extends the one given in [16, Definition 9] to take into account the sets of accepting states.

For notational convenience, given $\Sigma', \Sigma_1, \Sigma_2, \Sigma'' \subseteq \Sigma$, we define

$$\Sigma'\ {}_{\Sigma_1}\!\bigotimes_{\Sigma_2} \Sigma'' := (\Sigma' \cap \Sigma'') \cup (\Sigma' \cap \Sigma_1) \cup (\Sigma'' \cap \Sigma_2).$$

Given *generated* trajectory sets P, Q, the PSC of a pair of trajectories $e_p \in P$, $e_q \in Q$ was defined in [16, Definition 10] as follows:

Definition 3 Let P, Q be generated trajectory sets with $e_p \in P$, $e_q \in Q$. Then the PSC of e_p and e_q (with respect to P and Q), denoted $e_p\ {}_A\|_B\ e_q$, is defined inductively on $|e_p| + |e_q|$ as follows:

$\forall \Sigma_p, \Sigma_q \subseteq \Sigma$ such that $\Sigma_p \in P, \Sigma_q \in Q$:

$$\Sigma_p\ {}_A\|_B\ \Sigma_q := \{\Sigma' \subseteq \Sigma_p\ {}_A\!\bigotimes_B \Sigma_q\},$$

$\forall e_p \in P; e_q \in Q; \sigma_p, \sigma_q \in \Sigma; \Sigma_p, \Sigma_q \subseteq \Sigma$ such that $e_p(\sigma_p, \Sigma_p) \in P, e_q(\sigma_q, \Sigma_q) \in Q$:

$$e_p(\sigma_p, \Sigma_p)\ {}_A\|_B\ e_q(\sigma_q, \Sigma_q) := T_1 \cup T_2 \cup T_3,$$

where

$$T_1 := \begin{cases} \{e(\sigma_p, \Sigma') \mid e \in e_p\ {}_A\|_B\ e_q(\sigma_q, \Sigma_q); \Sigma' \subseteq \Sigma_p\ {}_A\!\bigotimes_B \Sigma_q\} & \text{if } \sigma_p \notin B \text{ and} \\ & e_q(\sigma_q, \Sigma_q)(\sigma_p, \emptyset) \notin Q \\ \emptyset & \text{otherwise} \end{cases}$$

$$T_2 := \begin{cases} \{e(\sigma_q, \Sigma') \mid e \in e_p(\sigma_p, \Sigma_p)\ {}_A\|_B\ e_q; \Sigma' \subseteq \Sigma_p\ {}_A\!\bigotimes_B \Sigma_q\} & \text{if } \sigma_q \notin A \text{ and} \\ & e_p(\sigma_p, \Sigma_p)(\sigma_q, \emptyset) \notin P \\ \emptyset & \text{otherwise} \end{cases}$$

$$T_3 := \begin{cases} \{e(\sigma, \Sigma') \mid e \in e_p\ {}_A\|_B\ e_q; \Sigma' \subseteq \Sigma_p\ {}_A\!\bigotimes_B \Sigma_q\} & \text{if } \sigma_p = \sigma_q := \sigma \\ \emptyset & \text{otherwise} \end{cases}$$

It should be noted that $e_p\ {}_A\|_B\ e_q$ is a set of refusal-traces that depends on the generated trajectory sets P, Q as well as on the particular trajectories e_p, e_q. The dependence on P, Q is not explicitly indicated in the notation.

Using the definition of PSC of refusal-traces, we next define the PSC of trajectory models.

Definition 4 Let (P^m, P), (Q^m, Q) be trajectory models with priority sets $A, B \subseteq \Sigma$ respectively. The PSC of (P^m, P) and (Q^m, Q), denoted $(P^m, P)\ {}_A\|_B\ (Q^m, Q)$, is the pair of refusal-trace sets (R^m, R), where

$$R^m := \bigcup_{e_p \in P^m, e_q \in Q^m} e_p\ {}_A\|_B\ e_q, \quad R := \bigcup_{e_p \in P, e_q \in Q} e_p\ {}_A\|_B\ e_q,$$

where $e_p\ {}_A\|_B\ e_q$ is with respect to P, Q in the definitions of both R^m and R.

We will use the notation $(P^m\ {}_A\|_B\ Q^m, P\ {}_A\|_B\ Q)$ for $(P^m, P)\ {}_A\|_B\ (Q^m, Q)$. However, it must be kept in mind that $P^m\ {}_A\|_B\ Q^m$ implicitly depends on P and Q.

Theorem 2 [11] $T^m(\mathcal{P})\ {}_A\|_B\ T^m(\mathcal{Q}) = T^m(\mathcal{P}\ {}_A\|_B\ \mathcal{Q})$ and $T(\mathcal{P})\ {}_A\|_B\ T(\mathcal{Q}) = T(\mathcal{P}\ {}_A\|_B\ \mathcal{Q})$.

Corollary 1 Let $\mathcal{P}_1, \mathcal{P}_2, \mathcal{Q}_1, \mathcal{Q}_2$ be NSM's such that $T^m(\mathcal{P}_1) = T^m(\mathcal{P}_2), T(\mathcal{P}_1) = T(\mathcal{P}_2)$, $T^m(\mathcal{Q}_1) = T^m(\mathcal{Q}_2), T(\mathcal{Q}_1) = T(\mathcal{Q}_2)$. Then for any $A, B \subseteq \Sigma$:

1. $T^m(\mathcal{P}_1\ {}_A\|_B\ \mathcal{Q}_1) = T^m(\mathcal{P}_2\ {}_A\|_B\ \mathcal{Q}_2)$ and $T(\mathcal{P}_1\ {}_A\|_B\ \mathcal{Q}_1) = T(\mathcal{P}_2\ {}_A\|_B\ \mathcal{Q}_2)$,
2. $L^m(\mathcal{P}_1\ {}_A\|_B\ \mathcal{Q}_1) = L^m(\mathcal{P}_2\ {}_A\|_B\ \mathcal{Q}_2)$ and $L(\mathcal{P}_1\ {}_A\|_B\ \mathcal{Q}_1) = L(\mathcal{P}_2\ {}_A\|_B\ \mathcal{Q}_2)$.

The first part of Corollary 1 states that the trajectory model contains enough detail to support the prioritized synchronous composition of NSM's. The second part of Corollary 1 states that the trajectory model serves as a *language congruence* [4] with respect to the operation of prioritized synchronous composition.

Corollary 2 [11] Let (P^m, Q) and (Q^m, Q) be two trajectory models, and $A, B \subseteq \Sigma$. Then $(P^m\ {}_A\|_B\ Q^m, P\ {}_A\|_B\ Q)$ is a trajectory model.

The following corollary states that when the priority sets of two systems are each equal to the entire event set, then the language model of the composed system can be obtained as the intersection of the individual language models.

Corollary 3 [11] Let $\mathcal{P} := (X_{\mathcal{P}}, \Sigma, \delta_{\mathcal{P}}, x^0_{\mathcal{P}}, X^m_{\mathcal{P}})$, $\mathcal{Q} := (X_{\mathcal{Q}}, \Sigma, \delta_{\mathcal{Q}}, x^0_{\mathcal{Q}}, X^m_{\mathcal{Q}})$ be NSM's. Then

1. $L^m(\mathcal{P}\ {}_\Sigma\|_\Sigma\ \mathcal{Q}) = L^m(\mathcal{P}) \cap L^m(\mathcal{Q})$,
2. $L(\mathcal{P}\ {}_\Sigma\|_\Sigma\ \mathcal{Q}) = L(\mathcal{P}) \cap L(\mathcal{Q})$.

Corollary 4 [11] Let (P^m, P) and (Q^m, Q) be trajectory models. Then

1. $L(P^m\ {}_\Sigma\|_\Sigma\ Q^m) = L(P^m) \cap L(Q^m)$,
2. $L(P\ {}_\Sigma\|_\Sigma\ Q) = L(P) \cap L(Q)$.

Next we prove the *associative* property of PSC. It is immediate from Definition 2 that given NSM's $\mathcal{P}, \mathcal{Q}$ with priority sets $A, B \subseteq \Sigma$ respectively, the following holds for for each $x_r = (x_p, x_q) \in X_{\mathcal{R}}$, where $\mathcal{R} := \mathcal{P}\ {}_A\|_B\ \mathcal{Q}$:

$$\mathfrak{R}_{\mathcal{R}}(x_r) = \mathfrak{R}_{\mathcal{P}}(x_p)\ {}_A\!\otimes_B\ \mathfrak{R}_{\mathcal{Q}}(x_q) = [\mathfrak{R}_{\mathcal{P}}(x_p) \cap \mathfrak{R}_{\mathcal{Q}}(x_q)] \cup [\mathfrak{R}_{\mathcal{P}}(x_p) \cap A] \cup [\mathfrak{R}_{\mathcal{Q}}(x_q) \cap B].$$

Theorem 3 [11] Let $\mathcal{P}, \mathcal{Q}, \mathcal{R}$ be NSM's and $A, B, C \subseteq \Sigma$. Then

$$(\mathcal{P}\ {}_A\|_B\ \mathcal{Q})\ {}_{A\cup B}\|_C\ \mathcal{R} = \mathcal{P}\ {}_A\|_{B\cup C}\ (\mathcal{Q}\ {}_B\|_C\ \mathcal{R}).$$

Corollary 5 [11] Let (P^m, P), (Q^m, Q) and (R^m, R) be trajectory models and $A, B, C \subseteq \Sigma$. Then

1. $(P^m\ {}_A\|_B\ Q^m)\ {}_{A\cup B}\|_C\ R^m = P^m\ {}_A\|_{B\cup C}\ (Q^m\ {}_B\|_C\ R^m)$,
2. $(P\ {}_A\|_B\ Q)\ {}_{A\cup B}\|_C\ R = P\ {}_A\|_{B\cup C}\ (Q\ {}_B\|_C\ R)$.

Part 2 of Corollary 5 is stated without proof in [5]; a direct proof of part 2 that does not depend on the corresponding result for NSM's is given in [16, Theorem 3].

Next we define *augmentation*, which is the PSC of a given NSM with an NSM of the type $\mathcal{D} := (\{x^0_{\mathcal{D}}\}, \Sigma, \delta_{\mathcal{D}}, x^0_{\mathcal{D}}, \{x^0_{\mathcal{D}}\})$, the transition function of which is defined as:

$$\forall \sigma \in \Sigma \cup \{\epsilon\}, \quad \delta_{\mathcal{D}}(x^0_{\mathcal{D}}, \sigma) := \begin{cases} \{x^0_{\mathcal{D}}\} & \text{if } \sigma \in D \\ \emptyset & \text{otherwise,} \end{cases}$$

where $D \subseteq \Sigma$ is a given set of events. I.e., $\mathcal{D}$ is a deterministic state machine consisting of a single state having self-loops on events in $D \subseteq \Sigma$. It is clear that $L^m(\mathcal{D}) = L(\mathcal{D}) = D^*$ and $T^m(\mathcal{D}) = T(\mathcal{D}) = det(D^*)$.

Definition 5 Given an NSM $\mathcal{P} := (X_\mathcal{P}, \Sigma, \delta_\mathcal{P}, x^0_\mathcal{P}, X^m_\mathcal{P})$ and $D \subseteq \Sigma$, the *augmented NSM with respect to* D, denoted $\mathcal{P}^D$, is defined to be $\mathcal{P}^D := \mathcal{P}\ {}_\emptyset\|_\emptyset\ \mathcal{D}$. Given a trajectory model (P^m, P), *the augmented trajectory model with respect to* D, denoted $(P^m, P)^D$, is defined to be $(P^m, P)^D := (P^m, P)\ {}_\emptyset\|_\emptyset\ (det(D^*), det(D^*))$.

Using $((P^m)^D, P^D)$ to denote $(P^m, P)^D$, we have $(P^m)^D = P^m\ {}_\emptyset\|_\emptyset\ det(D^*)$ and $P^D = P\ {}_\emptyset\|_\emptyset\ det(D^*)$. As is always the case with the PSC of *recognized* trajectory sets, the expression for $(P^m)^D$ implicitly depends on the corresponding *generated* trajectory sets P and $det(D^*)$. Clearly the trajectory model $(T^m(\mathcal{P}^D), T(\mathcal{P}^D))$ of the augmented NSM $\mathcal{P}^D$ is equal to $((T^m(\mathcal{P}))^D, (T(\mathcal{P}))^D)$. Also, since $det(D^*)$ can always execute every event in D and can never execute any event in $\Sigma - D$, it follows that given any trajectory model (P^m, P), any $A \subseteq \Sigma - D$, and any $B \subseteq D$

$$\begin{aligned}(P^m)^D &:= P^m\ {}_\emptyset\|_\emptyset\ det(D^*) = P^m\ {}_A\|_B\ det(D^*),\\ P^D &:= P\ {}_\emptyset\|_\emptyset\ det(D^*) = P\ {}_A\|_B\ det(D^*).\end{aligned}$$

Next we show that the PSC of two systems is equivalent to the strict synchronous composition (over the union of the two priority sets) of the associated augmented systems.

Proposition 2 [11] Let $(P^m, P), (Q^m, Q)$ be trajectory models, and $A, B \subseteq \Sigma$. Then

1. $P^m\ {}_A\|_B\ Q^m = (P^m)^{B-A}\ {}_{A\cup B}\|_B\ Q^m = (P^m)^{B-A}\ {}_{A\cup B}\|_{A\cup B}\ (Q^m)^{A-B}$,
2. $P\ {}_A\|_B\ Q = P^{B-A}\ {}_{A\cup B}\|_B\ Q = P^{B-A}\ {}_{A\cup B}\|_{A\cup B}\ Q^{A-B}$.

4 Supervisory Control Using PSC

In [16], we studied the supervisory control of nondeterministic systems in the setting of trajectory models and prioritized synchronization under the assumption that all refusal-traces of the plant are marked and the desired behavior is specified by a prefix-closed language; thus the issue of deadlock/blocking was not investigated. In this section, we generalize the results in [16] to include non-closed specifications and arbitrary marking of the plant refusal-traces. In addition, we consider *modular synthesis* of supervisors.

Since trajectory models contain sufficient detail to support the operation of prioritized synchronization, we use trajectory models, rather than NSM's, to represent a discrete event system. Unless otherwise specified, the trajectory model of the plant and that of the supervisor are denoted by (P^m, P) and (S^m, S), and the priority set of the plant and that of the supervisor are denoted by A and B respectively. In the setting of supervisory control, $A = \Sigma_u \cup \Sigma_c$, where $\Sigma_u, \Sigma_c \subseteq \Sigma$ denote the sets of *uncontrollable* and *controllable* events respectively [14, 15]; and $\Sigma_d \subseteq B \subseteq \Sigma_c \cup \Sigma_d$, where $\Sigma_d \subseteq \Sigma$ denotes the set of so-called *driven* [4] or *forcible* [3, 2] or *command* events [1]. The sets Σ_u, Σ_c and Σ_d are pairwise disjoint and exhaust the entire event set. Note that since $A = \Sigma_u \cup \Sigma_c$ and $\Sigma_d \subseteq B$, $A \cup B = \Sigma$.

The controlled (or closed-loop) system is $(P^m, P)\ {}_A\|_B\ (S^m, S)$. Since A contains every event that is either controllable or uncontrollable, such events cannot occur solely in the supervisor. This is consistent with the modeling assumption made in the Ramadge-Wonham theory [14] that these events originate in the plant. Since B contains every driven event, these events cannot occur solely in the plant. This corresponds to the modeling assumption that driven events originate in the supervisor and are executed synchronously by the plant whenever possible. If there is only one supervisor–i.e., the non-modular case–then the assumption that $\Sigma_d \subseteq B \subseteq \Sigma_c \cup \Sigma_d$ should be strengthened to $B = \Sigma_c \cup \Sigma_d$ to reflect the modeling assumption that the occurrence of controllable events requires the participation of the supervisor. However, the weaker assumption is useful when modular synthesis is considered, since it is not necessary to require that every subsystem in a modular supervisor participate in each controllable event.

In certain situations, a plant may consist of several "sub-plants" operating under prioritized synchrony. In such a case, if (P_i^m, P_i) is the trajectory model of the ith sub-plant, and $A_i \subseteq \Sigma$ is its priority set, then the trajectory model of the composed plant is given by $(P^m, P) := (\|_{A_i} P_i^m, \|_{A_i} P_i)$, where the notation $(\|_{A_i} P_i^m, \|_{A_i} P_i)$ is used to denote the PSC of sub-plants $\{(P_i^m, P_i)\}$. (Since PSC is associative (Corollary 5), $(\|_{A_i} P_i^m, \|_{A_i} P_i)$ is well defined, and it follows from Corollary 2 that it is a trajectory model.) The priority set of the composed plant is given by $A := \cup_i A_i$.

Definition 6 Given a plant (P^m, P) with priority set $A \subseteq \Sigma$, a supervisor, with trajectory model (S^m, S) and priority set $B \subseteq \Sigma$, is said to be *non-marking* if $S^m = S$; it is said to be *language model non-blocking* if $pr(L(P^m\ {}_A\|_B\ S^m)) = L(P\ {}_A\|_B\ S)$; and it is said to be *trajectory model non-blocking* if $pr(P^m\ {}_A\|_B\ S^m) = P\ {}_A\|_B\ S$.

Note that if a supervisor is trajectory model non-blocking, then it is also language model non-blocking. On the other hand, if a plant as well as a supervisor are deterministic, and the supervisor is language model non-blocking, then it is also trajectory model non-blocking.

Next we obtain a necessary and sufficient condition for the existence of a non-marking and language model non-blocking supervisor. This result is then used to obtain a necessary and sufficient condition for the existence of a non-marking and trajectory model non-blocking supervisor.

Theorem 4 [11] Let (P^m, P) be the trajectory model of a plant, $A, B \subseteq \Sigma$ with $A \cup B = \Sigma$, and $K^m \subseteq L((P^m)^{\Sigma-A})$ with $K^m \neq \emptyset$. Then there exists a non-marking and language model non-blocking supervisor with trajectory model (S, S) such that $L(P^m\, {}_A\|_B\, S) = K^m$ if and only if

Relative-closure: $pr(K^m) \cap L((P^m)^{\Sigma-A}) = K^m$
Controllability: $pr(K^m)(A - B) \cap L(P^{\Sigma-A}) \subseteq pr(K^m)$.

In this case, S can be chosen to be $det(pr(K^m))$.

Remark 1 In contrast to the standard conditions [14] for the existence of a non-marking and language model non-blocking supervisor in the absence of driven events, the controllability and relative-closure conditions given in Theorem 4 refer to the language model of the *augmented* plant. It is easily demonstrated [16, Example 2] that this language model depends on the *trajectory model* of the plant and generally cannot be determined if only the language model of the plant is known.

We will obtain necessary and sufficient conditions for the existence of a deterministic non-marking and *trajectory model non-blocking* supervisor that imposes a desired closed-loop recognized language. The following preliminary results are needed.

Lemma 1 [11] Let (P^m, P) be a trajectory model, $H = pr(H) \subseteq \Sigma^*$, and $K := L(P) \cap H$. Then

1. $P\, {}_\Sigma\|_\Sigma\, det(H) = P\, {}_\Sigma\|_\Sigma\, det(K)$,
2. $P^m\, {}_\Sigma\|_\Sigma\, det(H) = P^m\, {}_\Sigma\|_\Sigma\, det(K)$.

Proposition 3 [11] Let (P^m, P) be the trajectory model of a plant, $A, B \subseteq \Sigma$ with $A \cup B = \Sigma$, and $K^m \subseteq L((P^m)^{\Sigma-A})$ with $K^m \neq \emptyset$. If (S, S) is any non-marking language model non-blocking deterministic supervisor with $L(P^m\, {}_A\|_B\, S) = K^m$, then

$$P\, {}_A\|_B\, det(pr(K^m)) = P\, {}_A\|_B\, S, \tag{1}$$
$$P^m\, {}_A\|_B\, det(pr(K^m)) = P^m\, {}_A\|_B\, S. \tag{2}$$

Theorem 5 [11] Let (P^m, P) be the trajectory model of a plant, $A, B \subseteq \Sigma$ with $A \cup B = \Sigma$, and $K^m \subseteq L((P^m)^{\Sigma-A})$ with $K^m \neq \emptyset$. Then there exists a non-marking and trajectory model non-blocking deterministic supervisor with trajectory model (S, S) such that $L(P^m\, {}_A\|_B\, S) = K^m$ if and only if

Relative-closure: $pr(K^m) \cap L((P^m)^{\Sigma-A}) = K^m$
Controllability: $pr(K^m)(A - B) \cap L(P^{\Sigma-A}) \subseteq pr(K^m)$
Trajectory-closure: $P\, {}_A\|_B\, det(pr(K^m)) = pr[P^m\, {}_A\|_B\, det(pr(K^m))]$.

In this case, S can be chosen to be $det(pr(K^m))$.

Remark 2 The controllability condition of Theorem 4 is needed for the existence of a supervisor such that the closed-loop generated language equals the closure of the language to be recognized by the closed-loop system, i.e., $L(P\, {}_A\|_B\, S) = pr(K^m)$. The relative-closure condition is needed so that the corresponding non-marking supervisor (S, S) yields equality of closed-loop recognized language and the desired language, i.e., $L(P^m\, {}_A\|_B\, S) = K^m$. Clearly, this design automatically yields that the non-marking supervisor (S, S) is also language model non-blocking. The extra trajectory-closure condition of Theorem 5 is needed so that the non-marking supervisor (S, S) is also trajectory model non-blocking. Thus all three conditions of Theorem 5 are needed for the existence of a non-marking and trajectory model non-blocking supervisor so that the closed-loop recognized language is the desired one.

Next we show that our approach is also suitable for modular supervisory control. It was shown in [19] that such a design results in significant computational advantage provided the specification languages are *non-conflicting*.

Definition 7 Given $K_1^m, K_2^m \subseteq \Sigma^*$, K_1^m and K_2^m are said to be *non-conflicting* if $pr(K_1^m) \cap pr(K_2^m) = pr(K_1^m \cap K_2^m)$.

Since $pr(K_1^m \cap K_2^m) \subseteq pr(K_1^m) \cap pr(K_2^m)$ for any $K_1^m, K_2^m \subseteq \Sigma^*$, K_1^m and K_2^m being non-conflicting is equivalent to $pr(K_1^m) \cap pr(K_2^m) \subseteq pr(K_1^m \cap K_2^m)$.

Theorem 6 [11] Let (P^m, P) be the trajectory model of a plant, $A, B_1, B_2 \subseteq \Sigma$ be such that $A \cup B_1 = A \cup B_2 = \Sigma$, $K_1^m, K_2^m \subseteq L((P^m)^{\Sigma - A})$, and (S_1, S_1), (S_2, S_2) be the trajectory models of two non-marking and language model non-blocking supervisors such that $L(P^m \,{}_A\|_{B_1}\, S_1) = K_1^m$ and $L(P^m \,{}_A\|_{B_2}\, S_2) = K_2^m$.

1. The non-marking supervisor (S, S), where $S := S_1 \,{}_{B_1}\|_{B_2}\, S_2$, is such that
$$L(P^m \,{}_A\|_{B_1 \cup B_2}\, S) = K_1^m \cap K_2^m.$$
2. (S, S) is language model non-blocking if and only if K_1^m and K_2^m are non-conflicting.
3. If (S_1, S_1) and (S_2, S_2) are deterministic and trajectory model non-blocking, then (S, S) is deterministic, and is trajectory model non-blocking if and only if
$$\begin{aligned} &pr[P^m \,{}_A\|_{B_1 \cup B_2}\, [det(pr(K_1^m)) \,{}_{B_1}\|_{B_2}\, det(pr(K_2^m))]] \\ &\qquad = pr(P^m) \,{}_A\|_{B_1 \cup B_2}\, [det(pr(K_1^m)) \,{}_{B_1}\|_{B_2}\, det(pr(K_2^m))]. \end{aligned} \qquad (3)$$

Remark 3 Since it is assumed that $A \cup B_1 = A \cup B_2 = \Sigma$, it follows that $\Sigma_d \subseteq B_1 \cap B_2$–i.e., every driven event belongs to the priority sets of both of the modular supervisors. Consequently, for a driven event to occur in the closed-loop system, it must occur in both S_1, S_2, and will occur in P if possible. Although we have not explicitly made the assumption that $\Sigma_c \subseteq B_1 \cup B_2$, this is a natural requirement, since events in $\Sigma - (B_1 \cup B_2)$ cannot be prevented by S. However, there is certainly no need to require that $\Sigma_c \subseteq B_1 \cap B_2$. Thus, for a controllable event to occur in the closed-loop system, it must occur in P and in at least one of the supervisors.

5 Conclusion

In this paper, we have extended our earlier work on supervisory control of nondeterministic systems to include markings so that the issue of blocking can be investigated. Since language model non-blocking is inadequate for certain nondeterministic systems, the stronger requirement of trajectory model non-blocking is introduced. Necessary and sufficient conditions are obtained for the existence of language model non-blocking as well as trajectory model non-blocking supervisors that meet given language specifications.

These necessary and sufficient conditions require that the tests of controllability, relative-closure and trajectory-closure be performed. In the case where the languages involved are regular, one way to perform the test of controllability/relative-closure of the desired language with respect to the generated language of the augmented (nondeterministic) plant is to construct a language-equivalent deterministic system and apply a known test for controllability/relative-closure. However, it is possible to perform these tests without having to do such a nondeterministic to deterministic conversion. Hence algorithms of polynomial complexity (polynomial in the number of states of the plant NSM and number of states in the recognizer of the desired language) for testing controllability/relative-closure can be obtained. A test for the trajectory-closure condition can be obtained in terms of *coaccessibility* [7] of the NSM obtained by SSC of the plant NSM and the deterministic generator of the closure of the desired language specification. Thus the computational complexity of testing the trajectory-closure condition is also polynomial.

We have also addressed the issue of modular supervisory control and shown that our approach is suitable for such designs. It can be shown that the conditions for the existence of such designs are also polynomially testable.

References

[1] S. Balemi, G. J. Hoffmann, P. Gyugyi, H. W. Toi, and G. F. Franklin. Supervisory control of a rapid thermal multiprocessor. Technical Report ISL/GFF/91-1, Department of Electrical Engineering, Stanford University, Stanford, CA 94305, November 1991.

[2] B. A. Brandin and W. M. Wonham. Supervisory control of timed discrete event systems. Technical Report CSGR 9210, University of Toronto, Toronto, Canada, 1992.

[3] C. H. Golaszewski and P. J. Ramadge. Control of discrete event processes with forced events. In *Proceedings of 26th IEEE Conference on Decision and Control*, pages 247–251, Los Angeles, CA, 1987.

[4] M. Heymann. Concurrency and discrete event control. *IEEE Control Systems Magazine*, 10(4):103–112, 1990.

[5] M. Heymann and G. Meyer. Algebra of discrete event processes. Technical Report NASA 102848, NASA Ames Research Center, Moffett Field, CA, June 1991.

[6] C. A. R. Hoare. *Communicating Sequential Processes.* Prentice Hall, Inc., Englewood Cliffs, NJ, 1985.

[7] J. E. Hopcroft and J. D. Ullman. *Introduction to Automata Theory, Languages and Computation.* Addison-Wesley, Reading, MA, 1979.

[8] K. M. Inan. An algebraic approach to supervisory control. *Mathematics of Control, Signals and Systems*, 5:151–164, 1992.

[9] K.M. Inan and P.P. Varaiya. Algebras of discrete event models. *Proceedings of the IEEE*, 77(1):24–38, 1989.

[10] R. Kumar, V. K. Garg, and S. I. Marcus. On controllability and normality of discrete event dynamical systems. *Systems and Control Letters*, 17(3):157–168, 1991.

[11] R. Kumar and M. A. Shayman. Non-blocking supervisory control of nondeterministic systems via prioritized synchronization. Technical Report 93-58, Institute for Systems Research, University of Maryland at College Park, College Park, MD, 1993.

[12] R. Milner. *A Calculus of Communicating Systems.* Springer Verlag, 1980.

[13] I. Phillips. Refusal testing. *Theoretical Computer Science*, 50:241–284, 1987.

[14] P. J. Ramadge and W. M. Wonham. Supervisory control of a class of discrete event processes. *SIAM Journal of Control and Optimization*, 25(1):206–230, 1987.

[15] P. J. Ramadge and W. M. Wonham. The control of discrete event systems. *Proceedings of IEEE: Special Issue on Discrete Event Systems*, 77:81–98, 1989.

[16] M. Shayman and R. Kumar. Supervisory control of nondeterministic systems with driven events via prioritized synchronizationa and trajectory models. *SIAM Journal of Control and Optimization*, 1992. Accepted.

[17] M. A. Shayman and R. Kumar. Supervisory control of nondeterministic discrete event dynamical systems. In *Proceedings of 1993 IEEE Conference on Decision and Control*, San Antonio, TX, December 1993. To appear.

[18] M. A. Shayman and R. Kumar. Supervisory control of nondeterministic systems. In *Proceedings of 1993 Conference on Information Sciences and Systems*, pages 587–592, Johns Hopkins University, Baltimore, MD, March 1993.

[19] W. M. Wonham and P. J. Ramadge. Modular supervisory control of discrete event systems. *Mathematics of Control Signals and Systems*, 1(1):13–30, 1988.

SEQUENTIAL SCALAR QUANTIZATION OF VECTORS

Raja Balasubramanian
Xerox Webster Research Center, 800 Phillips Rd 128-29E, Webster, NY 14580.

Charles A. Bouman, Jan P. Allebach,
School of EE, Purdue University, West Lafayette, IN 47907.

ABSTRACT

We propose an efficient vector quantization (VQ) technique which we call sequential scalar quantization (SSQ). The scalar components of the vector are individually quantized in a sequence, with the quantization of each component utilizing conditional information from the quantization of previous components. Unlike conventional independent scalar quantization (ISQ), SSQ has the ability to exploit inter-data correlation. At the same time, since quantization is performed on scalar rather than vector variables, SSQ offers a significant computational advantage over conventional VQ techniques. In order to analyze the performance of SSQ, we appeal to asymptotic quantization theory where the codebook size is assumed to be large. Closed form expressions are derived for the quantizer mean squared error (MSE). These expressions are used to compare the asymptotic performance of SSQ with other VQ techniques. Theoretical and experimental results show that SSQ far outperforms ISQ with respect to MSE, while offering a considerable reduction in computation over conventional VQ at the expense of a moderate increase in MSE.

1. INTRODUCTION

Vector quantization (VQ) has found increasing use in data compression applications such as image and speech coding. The technique is motivated by the well known result from Shannon's rate distortion theory that superior performance can always be achieved by coding vectors rather than scalars. As with scalar quantization, the objective in VQ design is to minimize a distortion criterion such as mean squared error (MSE). An iterative error minimization algorithm developed by Lloyd for scalar quantization was extended to the vector case by Linde, Buzo, and Gray [1]. The iterative nature of this algorithm makes it computationally very intensive. A host of other suboptimal but computationally simpler VQ techniques have been reported in the literature [2, 3]. Until recently, the general class of VQ techniques has not received much attention for practical implementation because of the high computational cost, and the lack of codebook structures that are amenable to hardware implementation.

In this paper, we propose a VQ technique, which we call sequential scalar quantization (SSQ). Although similar ideas have been alluded to in the literature [2, 3], to our knowledge, this method has not been theoretically analyzed or seriously pursued in any application. As the name implies, the idea behind the technique is to sequentially quantize the scalar components of a vector. The main computational savings arises from the fact that we are quantizing along scalar rather than vector dimensions. At the same time, due to its sequential nature, SSQ possesses the ability to exploit the correlation between the scalar components of a vector. In order to analyze the performance of SSQ with respect to a distortion measure, we appeal to asymptotic quantization theory, where the number of output quantization levels is assumed to be very large. Using this theory, we derive closed form expressions for the quantizer distortion as a function of the design parameters. While we will present the important results in this paper, the need for brevity has forced us to omit some of the detailed derivations. The reader is referred to [4] for a more complete analysis of asymptotic theory of SSQ.

While this paper focusses on a theoretical analysis of SSQ, we have also successfully applied the analysis to a practical problem, namely color image quantization, where a high quality color image is to be displayed on a device with a small palette of colors [5]. For this application, the asymptotic theory proves to be a useful tool for designing a quantizer even when the number of output levels is small. The reader is referred to [5] for details of the application.

2. SCALAR QUANTIZATION OF VECTORS

We begin by defining the VQ problem and introducing some notation. Let X be a random vector in R^k with probability density function $p(x)$. An N point k-dimensional vector quantizer Q: $R^k \rightarrow R^k$ is a function whose domain is the set of all possible values of X and whose range is a set of N vectors $C = \{y_1,\dots,y_N\}$ called a codebook. Such a quantizer defines a partition $S = \{S_1,\dots,S_N\}$ of N regions in R^k, where $S_i = \{x \in R^k : Q(x) = y_i\}$. The quantization consists of two steps: the codebook design, which involves an appropriate selection of the output vectors $y_1,\dots,y_N$; and the mapping of each input vector to one of the output vectors according to the rule $Q(x) = y_i$ if $x \in S_i$. Define a distortion measure $d(x,y)$, d: $R^k \times R^k \rightarrow [0, \infty)$. The quantizer is designed to minimize the expected distortion $D_k = E\{d(X,Q(X))\}$ between its input and output. The distortion measure that we will be using is the mean-squared error (MSE),

$$D_k = \frac{1}{k} E\left\{\|X - Q(X)\|^2\right\}, \tag{1}$$

where $\| \ \|$ denotes Euclidean distance. The two necessary conditions for a quantizer to be optimal with respect to the MSE distortion are that (i) the y_i are chosen to be the centroid of x in S_i, *i.e.* $y_i = E\{X \mid x \in S_i\}$; and (ii) each input x is quantized to one of the y_i's according to a nearest neighbor rule, *i.e.* $S_i = \{x \in R^k : \|x - y_i\|^2 \leq \|x - y_j\|^2, j = 1,\dots,N\}$ [3]. These two conditions give rise to an iterative optimization problem that introduces considerable computation into the codebook design.

Another suboptimal but computationally simpler approach to quantize a vector $X = [X_1,\dots,X_k]^t$ is to quantize each of its individual scalar components X_i, $1 \leq i \leq k$. This may be done either independently or in a sequential fashion. With independent scalar quantization (ISQ), the codebook C_i of scalar outputs is designed independently for each scalar component X_i, $1 \leq i \leq k$, according to its marginal distribution $p(x_i)$. The final codebook is a k-fold Cartesian product of the k scalar codebooks,. A 2-D example of this scheme is shown in Fig. 1a for a rotated uniform distribution. The symbols **x** denote output vectors, which are taken to be the centroids within each region. Vector mapping may be accomplished by independently encoding each X_i to a channel symbol through a set of k lookup tables (LUT's). The outputs of these LUT's are then independently decoded to the scalar outputs $Y_1,\dots,Y_k$, which constitute the output vector $Y = [Y_1,\dots,Y_k]^t = Q(X)$. Since the codebook design only involves quantization of scalar variables, ISQ involves far less computation than conventional VQ. However, since the components are quantized independently, no inter-data correlation is taken into account, and many output vectors are wasted outside the region of input distribution, as is seen in Fig. 1a. Despite its shortcomings in performance, ISQ has been a popular alternative to VQ due to its simplicity.

With sequential scalar quantization (SSQ), the first scalar X_1 is quantized to some predetermined number of levels N_1 based on its marginal distribution $p(x_1)$. Each subsequent X_i, $2 \leq i \leq k$, is then quantized based on a set of conditional distributions of X_i within regions formed from the quantization of the scalars $X_1,\dots,X_{i-1}$. A 2-D example of SSQ is shown in Fig. 1b for the same uniform distribution. In this example, first X_1 is quantized to N_1=5 levels. This results in a partition of intervals B_{1j}, $1 \leq j \leq 5$, in R, or columns B_{2j} in R^2. Next, we quantize X_2 but confine the quantization to the columns formed from the quantization of X_1. This results in a 2-D quantizer with N_2=18 output vectors in R^2. The encoding of input vectors may be performed through a sequential or multistage LUT, as shown in Fig. 2 for 3-D vectors. The input to the first LUT is X_1. The output symbol b_{i-1} of the $(i-1)$th LUT, $2 \leq i \leq k$, is then fed to the input of the i-th LUT along with the i-th scalar component X_i. Finally, the output symbol b_k of the last encoder is decoded to one of the output vectors in the codebook C. As is the case with ISQ, the SSQ codebook design only involves scalar quantization, and the encoding operation entails no computation. However, unlike ISQ, which uses only marginal input statistics, SSQ uses both marginal and conditional (or effectively the joint) input statistics, and thus has the ability to exploit inter-data correlation. Hence, we see in Fig. 1b that SSQ places its all output vectors within the region of support of the input distribution, thus requiring fewer output codevectors than ISQ to achieve the same quality level.

3. ASYMPTOTIC QUANTIZATION THEORY

In general, it is not possible to derive analytical expressions for the quantizer MSE. However, if the number of output quantization levels N is allowed to become asymptotically large, then it is possible to arrive at approximate closed form error expressions [3]. The asymptotic analysis not only provides intuition about the behavior of the quantizer, but also can serve as a valuable guide in the design of the quantizer even for small N. In the section below, we briefly outline basic asymptotic results that have been developed for scalar and vector quantizers. The reader is referred to [3, 6, 7] for details.

Let X be a random variable with distribution $p(x)$. Consider a 1-D quantizer Q: R → R with N output points. Let $N(x)dx$ be the number of quantization levels that lie in the interval $[x, x+dx]$. We define the quantizer density function $\lambda(x)$ as

$$\lambda(x) = \lim_{N\to\infty} \frac{N(x)}{N} \, . \tag{2}$$

Thus for sufficiently large N, the quantity $N\lambda(x)dx$ is approximately the number of quantization levels in the interval $[x, x+dx]$. The MSE of the quantizer may be approximated by Bennett's distortion integral [3]

$$D = E\{[X - Y]^2\} \approx \frac{1}{12N^2} \int \frac{p(x)}{\lambda(x)^2} dx \, . \tag{3}$$

where $Y = Q(X)$ is the quantizer output. It may be easily shown that the quantizer density function $\lambda(\cdot)$ that minimizes D is given by

$$\lambda(x) = \frac{p(x)^{1/3}}{\int p(x)^{1/3} dx} \, , \tag{4}$$

and the resulting minimum distortion is given by

$$D \approx \frac{1}{12N^2} \|p(x)\|_{1/3} \, , \tag{5}$$

where we have used the notation

$$\|p(x)\|_m \equiv \left[\int p(x)^m dx\right]^{1/m} \, . \tag{6}$$

If we wish to find the conditional MSE $E\{[X - Y]^2 \mid A\}$ given the event A, we simply replace the marginal distribution $p(x)$ in (3) - (6) with the conditional distribution $p(x \mid A)$.

Extending the analysis to k-dimensional vectors, let $\boldsymbol{X}$ be a random vector in R^k with distribution $p(\boldsymbol{x})$; p: $R^k \to R$, and let Q be an N point quantizer; Q: $R^k \to R^k$. We may define the quantizer density function $\lambda(\boldsymbol{x})$; λ: $R^k \to R$, in a manner analogous to (2), so that $N\lambda(\boldsymbol{x})d\boldsymbol{x}$ is the number of quantization levels in an incremental volume $d\boldsymbol{x}$ around $\boldsymbol{x}$. Na and Neuhoff [7] extend Bennett's distortion integral (3) to the vector case yielding

$$D_k \approx \frac{1}{N^{2/k}} \int \frac{m_k(x)}{\lambda(x)^{2/k}} p(x)dx \quad , \tag{7}$$

where D_k is the MSE as defined in (1), and $m_k(x)$ is a function characterized by the shapes of the cells in the vicinity of x, known as the inertial profile function. Gersho [6] conjectures that for large N, the optimal quantizer is one whose quantization cells are tessellations of a congruent polytope. This result implies that $m_k(x) = M_k$ is a constant with respect to x and may be taken outside the integral in (7). As with the scalar case, (7) can be optimized with respect to $\lambda(x)$, yielding the minimum distortion

$$D_k \approx \frac{M_k}{N^{2/k}} \|p(x)\|_{k/(k+2)}, \tag{8}$$

where $\|p(x)\|_m$ is defined in a manner analogous to (6). Equation (8) serves as a lower bound on MSE performance for an N point VQ in k-dimensional space.

4. ASYMPTOTIC ANALYSIS OF SSQ

We now analyze the asymptotic behavior of SSQ in a manner similar to that used to derive the results in Sec. 3. Since SSQ is fundamentally different from either ISQ or conventional VQ, we cannot simply use the results of Sec. 3. Rather, we will rederive the theory for the sequential structure using the basic results as building blocks. We begin in Sec. 4.1 by precisely defining the SSQ structure and making explicit the variables that we are free to choose in the design of the quantizer.

4.1 The SSQ Design Problem

Given an input random vector $X = [X_1,...,X_k]^t$ in R^k with distribution $p(x)$, we wish to design a codebook of N output vectors $C = \{y_1,...,y_N\}$ by sequentially quantizing the scalar components $X_1,...,X_k$. To simplify the notation, we will assume that the components are quantized in the order $X_1, X_2,..., X_k$. The quantization of a scalar coordinate to a fixed number of levels requires some rule Φ to place the decision and reconstruction levels along that coordinate. This rule will be treated as a design variable.

We begin by designing an N_1 level quantizer for X_1 using the marginal distribution $p(x_1)$. Here, N_1 is a design variable. This quantization creates a partition $P'_1 = \{B'_{11},..., B'_{1N_1}\}$ of intervals in R, or a partition $P_2 = \{B_{21},..., B_{2N_1}\}$ of cylinder sets in R^2. We denote by B'_{1j} the j-th quantization region in R, and by B_{2j} the cylinder set in R^2 given by $B_{2j} = B'_{1j} \times \mathrm{R}$. Next, we design a quantizer for X_2, which creates a refinement $P'_2 = \{B'_{21},..., B'_{2N_2}\}$ of P_2 containing N_2 quantization regions. Here N_2 is again a design variable. The refinement is obtained by quantizing X_2 within each B_{2j} to some predetermined number of levels n_{2j} based on its conditional distribution $p(x_2 \mid B_{2j})$. Referring to Fig 1b, we have $n_{21} = 3$, $n_{22} = 4$, *etc.* Note that the n_{2j}'s must sum to N_2.

We may generalize this discussion to quantization of the i-th scalar component X_i, $2 \leq i \leq k$. The result of quantization along the dimension x_{i-1} is a partition P'_{i-1} of N_{i-1} regions in R^{i-1}, or a partition P_i of N_{i-1} cylinder sets in R^i. The i-th quantizer produces a refinement P'_i of P_i with N_i regions, where $N_i \geq N_{i-1}$. This is accomplished by quantizing along x_i within each of the cylinder sets B_{ij}, $1 \leq j \leq N_{i-1}$, to n_{ij} levels according to the conditional distributions $p(x_i \mid B_{ij})$. The N_i's are design variables that must satisfy the constraint $1 \leq N_1 \leq N_2 \leq ... \leq N_k = N$, where N is the desired number of codewords in R^k. Furthermore, for each i, $2 \leq i \leq k$, the n_{ij}'s are also design variables that must satisfy the constraint

$$\sum_{j=1}^{N_{i-1}} n_{ij} = N_i \quad . \tag{9}$$

Our objective is to pick the design variables N_i, n_{ij}, and the quantization rule Φ along each coordinate x_i to minimize the overall MSE D_k given in (1) for a fixed codebook size $N_k = N$. Since the MSE is a separable distortion measure, we may write

$$D_k = \frac{1}{k}\sum_{i=1}^{k} d_i \ , \tag{10}$$

where $d_i = E\{[X_i - Y_i]^2\}$ is the MSE along x_i resulting from the quantization $Y_i = Q(X_i)$. In the following section, we will obtain approximate analytical expressions for the d_i's in the asymptotic case.

4.2 Asymptotic Optimization of SSQ

We now derive an asymptotic theory for SSQ that provides us with closed form expressions for the MSE of the quantizer in terms of the aforementioned design variables. This allows the optimization of the quantizer with respect to these variables. For the subsequent analysis, we make the following assumptions: 1) the number of quantization levels N_i, $1 \le i \le k$, and the relative allocations n_{ij} are large; 2) the probability distribution $p(x)$ of the source is relatively smooth; 3) the rule Φ for placing decision and reconstruction levels along a scalar dimension x is completely specified by the quantizer density function $\lambda(x)$; 4) quantization along x_i will attempt to minimize only the MSE d_i along that dimension, and in particular will not affect the MSE's $d_1,\ldots,d_{i-1}$ due to previous quantizations; 5) the probability of quantizer overload is assumed to be negligible [3].

We will derive the error expressions for the case where X is a 3-D vector. The results generalize to k dimensions in a straightforward manner. We begin by quantizing X_1 to N_1 levels using the marginal distribution $p(x_1)$. We obtain from (3) the asymptotic approximation to the MSE along x_1

$$d_1 \approx \frac{1}{12N_1^2}\int \frac{p(x_1)}{\lambda(x_1)^2}dx_1 \quad . \tag{11}$$

The optimal quantizer spacing $\lambda(x_1)$ that minimizes d_1 is given by (4) with $p(x)$ replaced by $p(x_1)$, and from (5), the resulting minimum d_1 is

$$d_1 \approx \frac{1}{12N_1^2}\|p(x_1)\|_{1/3} \quad . \tag{12}$$

Next, we derive an expression for d_2, the MSE along x_2, by conditioning on the cylinder sets B_{2j} formed from quantizing X_1

$$d_2 = E\{[X_2 - Y_2]^2\} = \sum_{j=1}^{N_1} E\{[X_2 - Y_2]^2 \mid B_{2j}\}P(B_{2j}) \tag{13}$$

where for brevity, we have used $P(B_{2j})$ to denote $P([X_1, X_2]^t \in B_{2j})$. Note that within each B_{2j}, the random variable X_2 has conditional distribution $p(x_2 \mid B_{2j})$ and is quantized to n_{2j} levels. We may use (3) to approximate the conditional distortion within each B_{2j} for large n_{2j}

$$E\{[X_2 - Y_2]^2 \mid B_{2j}\} \approx \frac{1}{12n_{2j}^2}\int \frac{p(x_2 \mid B_{2j})}{\lambda(x_2 \mid B_{2j})^2}dx_2 \ , \tag{14}$$

where $\lambda(x_2 \mid B_{2j})$ denotes the relative density of quantization levels along x_2 within B_{2j}. We know from (4) that the optimal $\lambda(x_2 \mid B_{2j})$ that minimizes (14) is

$$\lambda(x_2 \mid B_{2j}) = \frac{p(x_2 \mid B_{2j})^{1/3}}{\int p(x_2 \mid B_{2j})^{1/3}dx_2} \quad . \tag{15}$$

Substituting (15) into (14), we get a minimum error expression similar to (5), which can then be substituted into (13) to yield

$$d_2 \approx \frac{1}{12} \sum_{j=1}^{N_1} \frac{1}{n_{2j}^2} \left\| p(x_2 \mid B_{2j}) \right\|_{1/3} P(B_{2j}) \,. \tag{16}$$

Now we are left with the relative allocations n_{2j}, $1 \le j \le N_1$ which may be chosen to minimize (16) subject to the constraint (9) that they sum to N_2. This is a straightforward constrained minimization problem that is easily solved by a Lagrangian method to yield the optimal allocations

$$n_{2j} = N_2 \frac{\left[\left\| p(x_2 \mid B_{2j}) \right\|_{1/3} P(B_{2j}) \right]^{1/3}}{\sum_{l=1}^{N_1} \left[\left\| p(x_2 \mid B_{2l}) \right\|_{1/3} P(B_{2l}) \right]^{1/3}} \,. \tag{17}$$

The resulting minimum d_2 is given by

$$d_2 \approx \frac{1}{12 N_2^2} \left\{ \sum_{j=1}^{N_1} \left[\left\| p(x_2 \mid B_{2j}) \right\|_{1/3} P(B_{2j}) \right]^{1/3} \right\}^3 . \tag{18}$$

For large N_1, we may use the smoothness restriction on $p(x)$ to rewrite (18) as a Riemann sum and approximate it by an integral [4]:

$$d_2 \approx \frac{N_1^2}{12 N_2^2} \left\{ \int \left[\left\| p(x_2 \mid x_1) \right\|_{1/3} p(x_1) \lambda^2(x_1) \right]^{1/3} dx_1 \right\}^3 , \tag{19a}$$

$$= \frac{N_1^2}{12 N_2^2} \left(\frac{\left\| \left\| p(x_2 \mid x_1) \right\|_{1/3} p(x_1)^{5/3} \right\|_{1/3}}{\left\{ \int p(x_1)^{1/3} dx_1 \right\}^2} \right) , \tag{19b}$$

where (19b) is obtained by substituting for $\lambda(x_1)$ from (4) into (19a).

We can carry through a similar analysis to obtain the MSE d_3 along x_3 (details may be found in [4]). Briefly, we condition on the cylinder sets B_{3j}, $1 \le j \le N_2$ formed from quantization of X_2

$$d_3 = E\{[X_3 - Y_3]^2\} = \sum_{j=1}^{N_2} E\{[X_3 - Y_3]^2 \mid B_{3j}\} P(B_{3j}) \,,$$

$$\approx \cdot \cdot \cdot \cdot \approx \frac{N_2^2}{12 N_3^2} \left(\frac{\left\| \left\| p(x_3 | x_1, x_2) \right\|_{1/3} p(x_1, x_2)^{5/3} p(x_1)^{4/9} \right\|_{1/3}}{\left\{ \int\int p(x_1, x_2)^{1/3} p(x_1)^{2/9} dx_1 dx_2 \right\}^2} \right) . \tag{20}$$

Upon inspection of (12), (19b), and (20), we see that we have factored each error term as the product of a constant and a term that depends only on the number of quantization levels. Substituting (12), (19b), and (20) into (10), we see that the overall 3-D MSE D_3 is of the form

$$D_3 \approx \frac{1}{36} \left(\frac{1}{N_1^2} \alpha_1 + \frac{N_1^2}{N_2^2} \alpha_2 + \frac{N_2^2}{N_3^2} \alpha_3 \right) , \tag{21}$$

where α_i are the constant terms associated with d_i, $1 \le i \le 3$. Recall that $N_3 = N$ is the known codebook size. If $\alpha_i > 0$, $i = 1, 2, 3$ (*i.e.* if there is a probabilistic variation along each x_i), then the expression (21) may be successively minimized with respect to N_1 and N_2 to yield the optimal allocations

$$N_1 = N_2^{1/2}\left(\frac{\alpha_1}{\alpha_2}\right)^{1/4} \quad ; \quad N_2 = N_3^{2/3}\left(\frac{\sqrt{\alpha_1\alpha_2}}{\alpha_3}\right)^{1/3} . \tag{22}$$

The resulting distortion is

$$D_3 \approx A_3 \frac{(\alpha_1\alpha_2\alpha_3)^{1/3}}{N_3^{2/3}} , \tag{23}$$

where $A_3 = 1/12$.

Generalizing to k-dimensional vectors, it may be shown using arguments similar to the one presented above, that the overall MSE D_k for an N point SSQ is given by

$$D_k \approx \frac{1}{12k}\sum_{i=1}^{k}\frac{N_{i-1}^2}{N_i^2}\alpha_i , \tag{24}$$

where $N_0 \equiv 1$ and $N_k = N$. The term α_i is a function of the conditional distribution $p(x_i \mid x_1,\ldots,x_{i-1})$ and all lower order distributions. If $\alpha_i > 0$, $1 \leq i \leq k$, then we may optimize D_k with respect to the N_i's and obtain expressions similar to (22), where each N_i is defined successively in terms of N_{i+1}. The resulting overall MSE is of the form

$$D_k \approx \frac{A_k}{N^{2/k}}\prod_{i=1}^{k}\alpha_i^{1/k} , \tag{25}$$

where A_k is a dimension dependent number. Comparing (25) with (8), the lower bound on the MSE that is achieved by the optimal vector quantizer, we see that asymptotically, the MSE for both SSQ and optimal VQ in k dimensions decreases as $1/N^{2/k}$. This implies that a fixed percentage loss in performance, independent of N, will be incurred by using SSQ instead of the optimal VQ method. The loss will depend on the relative values of the constants in (8) and (25), which in turn depend on the joint, conditional, and marginal statistics of the source X. The suboptimal performance of SSQ is explained by the rectangular cell shapes resulting from this technique, and the fact that SSQ may not achieve the optimal density $\lambda(\mathbf{x})$ of quantization points in k-dimensional space.

We note that the foregoing analysis may be readily extended to weighted MSE distortion measures and to any r-th power distortion measure for $r \geq 1$. The only restriction is that the measure must be separable along the scalar coordinates.

4.3 Analysis of SSQ for a Gaussian Source Distribution

The analysis above may be used to evaluate and compare the asymptotic performance of SSQ, ISQ, and VQ for the case where the input is a jointly Gaussian random vector. Let $X = [X_1, X_2]^t$ be characterized by the 2-D jointly Gaussian density $\mathrm{N}[\mu_1, \mu_2, \sigma_1, \sigma_2, \rho]$. We can assume, without loss of generality, that $\mu_1 = \mu_2 = 0$. Noting that $N_2 = N$ is fixed for the 2-D case, we may use the results obtained above to arrive at a simple expression for the 2-D MSE resulting from SSQ [4]

$$D_2^{SSQ} \approx \frac{0.777\pi}{N}\sigma_1\sigma_2\sqrt{1-\rho^2} . \tag{26}$$

We may compare (26) with the lower bound given in (8) for our example distribution. Since SSQ is restricted to rectangular cell shapes, we would obtain a more meaningful lower bound on its performance if we let $M_2 = 1/12$ in (8), which corresponds to VQ with rectangular cell shapes. This results in the lower bound:

$$D_2^{OPT} \approx \frac{0.667\pi}{N}\sigma_1\sigma_2\sqrt{1-\rho^2} . \tag{27}$$

Hence, in the asymptotic case, there is a 16.5% loss in using SSQ over a VQ scheme with square cells. Finally, if we assume that X_1 and X_2 are quantized independently according to their marginal distributions, then we obtain the following asymptotic error [2]

$$D_2^{ISQ} \approx \frac{0.886\pi}{N}\sigma_1\sigma_2. \tag{28}$$

As expected, the performance of ISQ does not depend on the correlation coefficient ρ. If we let $\rho = 0$ (*i.e.*, X_1 and X_2 are both uncorrelated and independent), and compare (26) and (28), we arrive at the interesting result that SSQ outperforms ISQ even when the components are independent!

5. EXPERIMENTAL RESULTS

5.1 Scalar Quantization According to λ(x)

Even though the notion of $\lambda(x)$. is only meaningful in an asymptotic sense, it is possible to use this function to design a low rate quantizer. In [5], we describe a rule to partition a scalar dimension into N intervals such that the distribution of quantization levels follows $\lambda(x)$. Namely, the quantization intervals are chosen so that the area under the function $\lambda(x)$ given in (4) within each interval is equal. We compared this method, which we call the lambda technique, with the scalar version of the binary splitting (BSP) algorithm [8] and the Lloyd-Max (LM) quantizer [3] for a simulated Gaussian source distribution. Since the Gaussian density is log concave, the LM algorithm yields a globally minimum solution and serves as a lower bound on quantizer performance. The lambda technique was used as a starting point for the LM iterations. In order to obtain a meaningful comparison, we normalized the MSE of each algorithm by that of the LM algorithm. For $N < 16$, the MSE due to the lambda technique is only about 5 % higher than the minimum MSE. As N increases, the lambda technique rapidly converges to the minimum MSE solution, verifying that this technique is indeed asymptotically optimal. The BSP algorithm results in an error that is consistently about 15% higher than the minimum MSE, suggesting that at least for 1-D Gaussian data, this is not an optimal strategy even in the asymptotic sense.

5.2 Verification of SSQ Analysis in 2-D

We performed quantization using SSQ on 2-D simulated Gaussian data and compared the resulting experimental MSE with the theoretical expression given in (26). Figure 3 shows a plot of the percentage difference between experimental and theoretical MSE as a function of the number of quantization levels N in 2 dimensions for two examples of jointly Gaussian distributions. We see that in both cases, as N becomes large, the experimental error converges to the formula (26), thus confirming the validity of the analysis in 2 dimensions.

5.3 Performance of SSQ on Experimental Data

We have compared the performance of ISQ, SSQ, and VQ for the quantization of 3-D color image data. Here, we will give only a brief verbal discussion of the results. The MSE performance of SSQ is far superior to that of ISQ, while BSP offers a moderate improvement over SSQ. The superior performance of SSQ over ISQ is a result of the exploitation by SSQ of correlation in the data; while the improved performance of BSP over SSQ is due mainly to the fact that the former is not constrained to rectangular cells. The qualitative performance of the 3 techniques is generally in corroboration with MSE performance.

We have also compared the computation times on a Sun SPARCStation 2 of ISQ, SSQ, and BSP on various 3-D image data sets. Let M and N be the image size and codebook size respectively. The complexities of both ISQ and SSQ are $O(M.)$, with that of SSQ being about 3 times that of ISQ. The complexity of BSP is $O(M \log_2 N)$ [8]. This dependence of computation on codebook size is characteristic of most VQ techniques. If r is the ratio of computation times of BSP to ISQ, then $r \approx 14$ for $N = 256$, and $r \approx 20$ for $N = 1024$.

8. CONCLUSIONS

We have proposed an efficient technique for quantizing vectors which we call sequential scalar quantization. A theoretical analysis of SSQ in the asymptotic case yields intuitive and useful results that allow us both to compare this method with other quantization strategies and to design a practical quantizer. We have theoretically and experimentally demonstrated the fact that SSQ yields considerably improved MSE performance over conventional independent scalar quantization, while offering a significant computational advantage over conventional VQ. Moreover the resulting sequential structure of this technique lends itself very easily to a hardware embodiment. While we have successfully applied SSQ to the color image quantization problem, this technique is of potential use in any vector quantization application where computational cost is an important consideration, and a moderate loss in quantitative performance can be tolerated.

9. ACKNOWLEDGEMENTS

This work was partially supported by an NEC Faculty Fellowship (C. A. B.) and an Eastman Kodak Company Fellowship (R. B.).

10. REFERENCES

[1] Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design," *IEEE Trans. Commun.*, vol. COM-28, pp. 84-95, Jan. 1980.

[2] J. Makhoul, S. Roucos, and H. Gish, "Vector quantization in speech coding," *Proc. IEEE*, 73, 1551-1588 (1985).

[3] A. Gersho, R. M. Gray, *Vector Quantization and Signal Compression,* Kluwer Academic Publishers, Norwell, 1991.

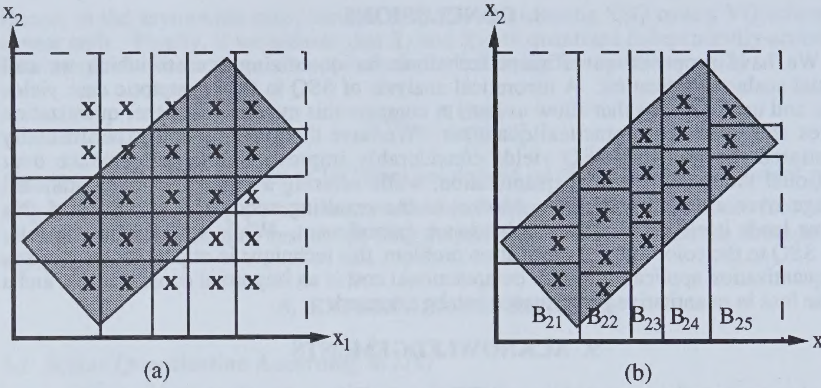
[4] R. Balasubramanian, C. A. Bouman, and J. P. Allebach, "Sequential scalar quantization of vectors: An analysis," submitted to *IEEE Trans. Image Processing.*

[5] R. Balasubramanian, C. A. Bouman and J. P. Allebach, "Sequential scalar quantization of color images," submitted to *Journal of Electronic Imaging.*

[6] A. Gersho, "Asymptotically optimal block quantization," *IEEE Trans. Inform. Thy.*, vol. IT-25, pp. 373-380, July 1979.

[7] S. Na and D. L. Neuhoff, "Bennett's integral for vector quantizers, and applications," *1990 IEEE Int'l Symposium on Information Theory*, Jan. 1990.

[8] R. Balasubramanian. C. A. Bouman and J. P. Allebach, "New results in color image quantization," *Proceedings of the 1992 SPIE/SPSE Symposium on Electronic Imaging - Science and Technology*, San Jose, CA, February 10 - 13, 1992.


x2
x1
X
B21
B22
B23
B24
B25

(a)

(b)

Fig. 1 2-D Example of (a) independent scalar quantization, and (b) sequential scalar quantization.

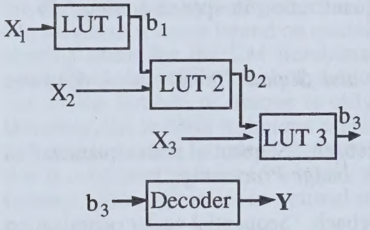

X1
LUT 1
b1
X2
LUT 2
b2
X3
LUT 3
b3
b3
Decoder
Y

Fig. 2 Encoder-Decoder operation in sequential scalar quantization

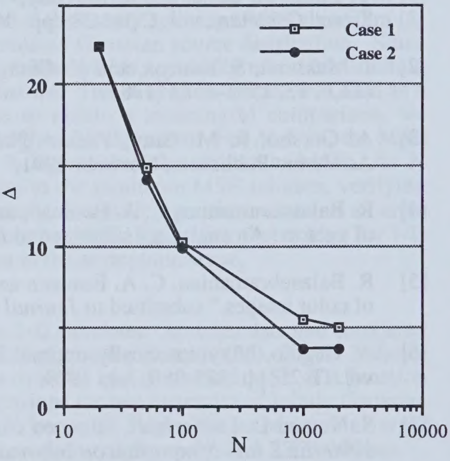


Fig. 3 Percentage discrepancy Δ between experimental and theoretical MSE's as a function of the codebook size N for 2-D Gaussian data (Case 1: $\sigma_1^2 = 1/3$, $\sigma_2^2 = 1/6$, $\rho = 0.8$; Case 2: $\sigma_1^2 = \sigma_2^2 = 1/4$, $\rho = 0$).

OPTIMAL BUFFER CONTROL FOR VARIABLE-RATE LOSSY COMPRESSION

David Tse, Robert Gallager and John Tsitsiklis
Laboratory for Information and Decision Systems, M.I.T.
email: dntse@lids.mit.edu

Abstract

In many lossy data compression systems, such as the one for HDTV, the data stream generated by the source coder has a variable rate, while the channel on which the data is to be transmitted operates in a fixed-rate manner. This necessitates the use of a buffer and a buffer control scheme whereby parameters of the compression system are adjusted based on the state of the buffer, to avoid overflow and at the same time to maintain adequate performance in terms of average distortion. We study a simple model and derive asymptotically optimal control scheme in the regime of large buffer size.

1 Introduction

Problems in rate-distortion theory are usually posed in terms of minimizing the average distortion in quantizing the source given a constraint in the *average* rate of the encoded bit stream. Very often, however, there are more stringent constraints on the behavior of the output process from a source coder. For example, there is usually a fixed bandwidth allocated for the broadcasting of compressed digital television signals, implying that the encoded bit stream must be fed into a fixed-rate channel. In data networks, a variable-rate output bit stream can usually be accommodated, but there are often rate-control schemes which regulate the burstiness of the stream by enforcing a peak rate as well as an average rate.

While it is theoretically possible to approach the rate-distortion limit using vector quantization and a fixed-length encoding of the reproduction vectors, the block length required of the vector quantizer is much too large to be useful in practice. With a small block length, variable-length encoding of the reproduction vectors can attain a significantly smaller distortion for a given average bit rate, by taking advantage of the statistical variability of the source. However, this necessitates the use of a buffer between the source coder and the channel in order to satisfy the constraints described above.

A general model is as follows. The input real-valued data are represented by the stochastic process $\mathbf{Y_1}, \mathbf{Y_2}, \ldots$, where $\mathbf{Y_t} \in \Re^k$ and k is the block length of the data to be quantized. The statistics of $\{\mathbf{Y_t}\}$ are assumed to be known and the block length k is pre-specified. At each time t, a quantizer Q is chosen which maps the vector $\mathbf{Y_t}$ into a variable-length binary codeword $\mathbf{C_t} = Q(\mathbf{Y_t})$ representing the reproduction vector into which $\mathbf{Y_t}$ is quantized. (Here, we are regarding Q as a composition of both the vector quantization and the lossless encoding of the reproduction vector.) The codeword $\mathbf{C_t}$ is then put into a buffer of size L. At each time instant t, a fixed number, R_c, of bits is taken out of the buffer and put into the channel. A *buffer control* scheme chooses the quantizer Q for the next source vector, based on the number of bits S_t currently in the buffer and the previous codewords $\mathbf{C_1}, \mathbf{C_2}, \ldots, \mathbf{C_t}$. (Assuming no transmission errors, this information is also available to the decoder so it can always keep track of the quantizer used at each time.) The problem is to find the buffer control scheme that minimizes the average distortion while keeping the buffer from overflowing.

While the above formulation naturally encompasses the case when the channel has a strictly fixed rate, it is also appropriate to model some situations where burstiness is allowed. One such scheme is the leaky bucket [3], which, when given two parameters R_c and W, implements a rate-control scheme such that the average rate of bits entering into the channel is R_c and the maximum number of bits going into the channel per unit time slot is W bits. One view of this scheme is to imagine that permits are generated at a rate of R_c at each time slot and are put into a bucket of size W. Excess permits are discarded. In time slot t, A_t data bits enter a buffer (of size L) and the ones at the bottom of the buffer can leave it and enter the channel as long as each bit can obtain a permit from the bucket. Thus, no more than W bits can enter the channel in any one time slot and the average rate is R_c. It is not difficult to show that if P_t is the number of permits in the bucket and S_t is the number of data bits waiting in the buffer at the beginning of time slot t, then $\tilde{S}_t$, defined by:

$$\tilde{S}_t = \begin{cases} S_t + W & \text{if } S_t > 0 \\ W - P_t & \text{if } S_t = 0 \end{cases}$$

is the state of a fictitious buffer of size $L + W$, with arrival process $\{A_t\}$ and a constant departure rate R_c bits per unit time. Moreover, since there is a one-to-one correspondence between the fictitious buffer state $\tilde{S}_t$ and the buffer-bucket state pair (S_t, P_t), any optimal control scheme for the fictitious buffer will translate into an optimal control scheme for the original leaky bucket problem. Thus, our formulation is general enough to include this case.

The problem of buffer overflow for *lossless* variable-length encoding has a relatively long history. It was first looked at by Jelinek [6], who gave conditions on codeword lengths that minimize the probability of overflow. This work was followed by Humblet [5] who presented an algorithm to actually compute the optimal code. Wyner [8] gave an approximate formula for the average fraction of data that will be lost due to overflow for any given variable-length lossless coding scheme. These results are all on lossless coding, where there is no tradeoff between average distortion and overflow probability and hence no opportunity for control. More recent and closer to the problem considered here is the work of Farvardin and Modestino [2] and Harrison and Modestino [4]. They presented specific buffer control schemes for variable-rate lossy coding of memoryless sources and computed numerically and via simulation their average distortion performance. No attempt was made to derive optimal schemes.

The objective of this paper is to gain a better understanding of the structure of *optimal* buffer control schemes. For a fixed source and buffer size L, the optimal control scheme can be obtained numerically as a solution to an average-cost dynamic programming problem. Such an approach, however, yields little insights to qualitative features of optimal schemes. Moreover, sources like video are very complex and a complete statistical characterization is often impossible. To get more theoretical insights, we shall rather consider the asymptotic problem when the buffer size L becomes large, and analytically derive optimal buffer control schemes for that regime. As a first step in this program, we will treat the case of memoryless sources. More complex sources will be considered in future papers.

In the next section, we will give a more precise formulation of the problem. Section 3 contains our results. We will present a lower bound on the achievable performance as well as a very simple control scheme that almost attains that lower bound. The performance is also compared against what is achievable without any control. Section 4 contains the conclusions. In this extended abstract, the proofs will only be sketched. Also, the following short-hand notations will be used to compare rates of convergence: $a_n = O(b_n)$ if a_n goes to zero at least as fast as b_n; $a_n = o(b_n)$ if a_n goes to zero strictly faster than b_n; $a_n = \Omega(b_n)$ if a_n goes to zero no faster than b_n; $a_n = \Theta(b_n)$ if they go to zero at the same rate.

2 Formal Problem Statement

Let $\mathbf{Y_1}, \mathbf{Y_2}, \ldots$ be an i.i.d. sequence of source samples, where $\mathbf{Y_t} \in \Re^k$ has a density p_y, which is assumed to be sufficiently smooth. . A quantizer $Q : \Re^k \to \{0,1\}^*$ is a composition of two maps, $Q = V \circ U$ where $U : \Re^k \to \Re^k$ maps the data sample to a reproduction vector, and $V : \Re^k \to \{0,1\}^*$ assigns a binary codeword to the reproduction vector. The lossless code V is assumed to be uniquely decodable and satisfy the prefix-condition. Two important attributes of a quantizer are its mean-square distortion and its rate:

$$D(Q) = E(\|\mathbf{Y} - U(\mathbf{Y})\|^2), \qquad R(Q) = E(|Q(\mathbf{Y})|)$$

(Here, $|\cdot|$ denotes the length of the codeword.)

To characterize the optimal distortion that can be achieved for a given *average* rate, define the operational distortion-rate curve as:

$$D_{op}(R) = \inf_{\{Q:E(|Q(\mathbf{Y})|)\leq R\}} D(Q)$$

This is the minimum distortion one can achieve using the same quantizer on each data vector, with only an average rate constraint. It is not true in general that this curve is convex. If it is not, then in fact one can do better by time-sharing between points on this curve, i.e. we can use two quantizers, each for a fraction of the time, and achieve a better average distortion than one single quantizer at the same average rate. Let us now define the time-shared distortion-rate curve, $D_T(R)$, as the lower convex hull of $D_{op}(R)$. This curve will be a union of strictly convex sections, on which the optimum can be attained by a single quantizer, and straight-line segments, on which time-sharing takes place. Points on $D_T(R)$ represent the minimum distortion that can be achieved for a given source block length and average rate, and without any buffer constraints. The problem is how well we can do relative to this optimum when there is a finite buffer.

Let us now turn to the setting with a finite buffer and an adaptive choice of quantizers. Since the data samples are i.i.d., it suffices to consider only memoryless buffer control schemes where the choice of the quantizer for the next sample is only a function of the buffer state and not of the previous quantized samples. Given a current buffer state $S_t^L = s$, let the quantizer selected for the next sample be Q_s^L. (The explicit dependence on the buffer size L is shown to emphasize that a different control scheme can be chosen for each L.) Let the net number of bits entering the buffer from the next sample be $X_{t+1}^{L,s} \equiv |Q_s^L(Y_{t+1})| - R_c$. If this results in an underflow of the buffer, i.e. $X_{t+1}^{L,s} + S_t^L < 0$, the deficit is simply padded with zeros and sent onto the channel. (Assuming the decoder also keeps track of the buffer state, there is no problem in decoding here.) On the other hand, if the current sample results in an overflow of the buffer, i.e. $X_{t+1}^{L,s} + S_t^L > L$, then the part of the codeword that fits into the buffer is sent, thus filling up the buffer. We assume that in decoding this incomplete codeword, a fixed and large distortion of D_0 is committed, where $D_0 = E(\|\mathbf{Y} - E(\mathbf{Y})\|^2)$. In other words, the decoder treats this as a garbled codeword and just decodes it to the mean vector. In practice, one can do something more sophisticated, such as sending the nearest codeword which will fit into the buffer [1]. But the specific overflow handling scheme will have little bearing on our asymptotic analysis, so in this paper we will concentrate on the simpler scheme.

Given the buffer size L and a specific buffer control scheme, the buffer state process $\{S_t^L\}$ forms a finite-state Markov chain. Assuming that the chain is ergodic, let its steady state distribution be π_L. Since a large distortion is committed whenever the buffer becomes full, the tradeoff between the steady-state probability the buffer is full, $p_f(L) \equiv \pi_L(L)$, and the steady state average distortion in normal operation when the buffer is not full, $D_q(L) \equiv \sum_s \pi_L(s) D(Q_s^L)$, is of interest. For reasonable control schemes, one would expect that $p_f(L) \to 0$ and $D_q(L) \to D(R_c)$ as the buffer size L becomes large, where $D_T(R_c)$ is the minimum distortion achievable with an average rate constraint of R_c. The key question that will be addressed is the optimal tradeoff between these two rates of convergence, and which control scheme can achieve this optimal tradeoff. Another way of viewing the problem is to consider the total distortion as the sum of the average distortion in normal operation and the large distortion when the buffer is full, ie. $D_q(L) + p_f(L) D_0$, and the control objective is to make both terms small.

3 Results

The key tool used to estimate the steady-state probability of a full buffer is Wald's identity [7], which yields information on hitting probabilities and hitting times of random walks.

Lemma 3.1 **(Wald's Identity** *) Let $\{W_n\}$ be a negative-drift random walk starting at the origin, and let $\Lambda(r) \equiv \log E(e^{rW_1})$ be its log moment generating function, which is assumed to be finite for all r. Let $a \geq 0$ and $b \leq 0$ be two given barriers, and let N be the first $n \geq 1$ such $W_n \geq a$ or $W_n \leq b$. Then*

$$E(e^{r^* W_N}) = 1$$

where r^* is the unique positive root of $\Lambda(r) = 0$. Furthermore, the expected hitting time is given by:

$$E(N)E(W_1) = E(W_N)$$

The tradeoff between the average distortion in normal operation and the fullness probability depends on whether the channel rate R_c lies in the time-shared regions of the distortion-rate curve or in the regions achievable by single quantizer. In the time-shared regions of the distortion-rate curve, one can achieve essentially optimal distortion with negligibly small probability of filling the buffer.

Proposition 3.2 *If $(R_c, D_T(R_c))$ lies on a straight-line time-shared segment of the distortion-rate curve, then there is a control scheme such that the average distortion $D_q(L)$ approaches $D_T(R_c)$ exponentially fast in L, and a fullness probability $p_f(L)$ decaying exponentially to zero with L.*

Sketch of Proof. Let (R_1, D_1) and (R_2, D_2) $(R_1 < R < R_2)$ be the two ends of the straight line segment containing $(R_c, D_T(R_c))$, and let Q_1 and Q_2 be the quantizers achieving these points respectively. Our scheme is as follows. For each L, use quantizer Q_2 when the buffer is less than half full and use Q_1 otherwise. Note that the selected quantizers are independent of the buffer size.

To compute $p_f(L)$, we view the times at which the buffer becomes full as the epochs of a renewal process and apply renewal theory to obtain $p_f(L) = \frac{1}{E(T)}$, where T is the duration between successive times when the buffer becomes full. Denote the upper and lower halves of the buffer by A_1 and A_2 respectively. To compute $E(T)$, we condition and decompose the evolution of the buffer state process with respect to events of barrier hitting, where the barriers are the top of the buffer, the center of the buffer, and the bottom of the buffer. Since in between these events, the process is simply a random walk, we can express $E(T)$ in terms of the expected barrier hitting times and hitting probabilities, which can in turn be estimated via Wald's identity. One finds that

$$\begin{aligned} E(T) &= \Theta\left(\frac{1}{P(\text{hitting the top before returning to center, starting from center})}\right) \\ &= \Theta(\exp(\frac{1}{2}r^*(Q_1)L)) \qquad \text{(by Wald's Identity)} \end{aligned}$$

where $r^*(Q_1)$ is the unique positive zero of the log moment generating function of the random variable $|Q_1(\mathbf{Y})| - R_c$.

We next compute the average distortion $D_q(L)$ for L large. This is given by:

$$D_q(L) = \pi_L(A_1)D_T(Q_1) + \pi_L(A_2)D_T(Q_2)$$

Since the buffer state process is bounded, the expected drift over its steady-state distribution is zero. Hence we can write the balance equation:

$$\pi_L(A_1)(R_1 - R_c) + \pi_L(A_2)(R_2 - R_c) + \text{ effects due to top and bottom of buffer } = 0$$

Since the fraction of time spent at the top and bottom of the buffer is exponentially small as L becomes large, it can be argued that the third term is also exponentially small, and

$$\begin{aligned} D_q(L) &= \frac{R_2 - R_c}{R_2 - R_1}D_T(Q_1) + \frac{R_c - R_1}{R_2 - R_1}D_T(Q_2) + \text{ exponentially small term} \\ &= D_T(R_c) + \text{ exponentially small term} \end{aligned}$$

□

Hence, if R_c is in the time-shared region, then both the average distortion in normal operation and the fullness probability approach their respective optimal values exponentially fast. Thus, there is really not much need to worry about the tradeoff between these two quantities. The reason for this nice situation is that one can use two fixed quantizers (for all buffer sizes) which, while achieving near-optimal average distortion during normal operation, also leads to very small probability of filling the buffer because of the large negative net drift of the quantizer used in the upper half of the buffer.

In the single-quantizer achievable region, on the other hand, there is a more stringent tradeoff involved. Since the distortion-rate curve is strictly convex at $R = R_c$, it is clear that the rates of the quantizers have to be close to R_c in order that their average distortion in normal operation can get close to the optimal distortion $D_T(R_c)$. The closer the average distortion is to the optimal, therefore, the smaller the negative drifts have to be, and the larger the chance of having a full buffer. And there lies the tradeoff.

We make two assumptions on the class of quantizers $\mathcal{Q}$ that are used in the buffer control schemes we shall be dealing with:

1) There exist $\epsilon > 0$ such that $P(|Q(Y)| > R_c) > \epsilon$ for all quantizers $Q \in \mathcal{Q}$.

2) There exists a uniform bound M on the codeword lengths for all $Q \in \mathcal{Q}$.[1]

The following result gives a lower bound on the tradeoff between the average distortion $D_q(L)$ and the fullness probability $p_f(L)$.

Proposition 3.3 *Assume that as the buffer size becomes large, the sequence of buffer control scheme satisfies:* $\lim_{L\to\infty} \max_s |\mu_{L,s}| = 0$, *where* $\mu_{L,s} \equiv E(|Q_s^L(\mathbf{Y})|) - R_c$ *is the net drift of the quantizer. Then the rates of convergence of the steady-state normal distortion* $D_q(L)$ *and of the fullness probability* $p_f(L)$ *are constrained as follows: if* $p_f(L) = o(1/L^2)$, *then* $D_q(L) - D_T(R_c) = \Omega(1/L^2)$.

In essence, the result says that no buffer control scheme can make both $p_f(L)$ and $D_q(L)$ converging at a rate faster than $1/L^2$. If we take the total average distortion as $D_q(L) + D_0 p_f(L)$, then the results implies that a lower bound on the convergence rate of the total distortion to the optimal distortion $D_T(R_c)$ is $1/L^2$.

Sketch of Proof. Here, we consider only buffer control schemes which have drifts symmetrically about the center of the buffer, i.e. $\mu_{L,\frac{L}{2}+s} = -\mu_{L,\frac{L}{2}-s}$ for $s = 0, 1, \ldots, \frac{L}{2}$. The proof for non-symmetrical schemes use similar ideas but require a more elaborate argument, and will not be sketched here.

---

[1] This assumption is stronger than necessary, and, with more technical work, can probably be replaced by a condition on the uniform boundedness on certain moments of $|Q(\mathbf{Y})|$.

The key idea is to write an appropriate balance equation for the drifts. Because the drifts are always pointed towards the buffer center, one can argue that the center is the most probable state, so that $\pi_L(\frac{L}{2}) = \Omega(\frac{1}{L})$. If we define a new Markov chain by grouping the entire lower half of the buffer into one state s^* and look at the steady-state balance equation for the new chain, we get:

$$\begin{aligned} 0 \;=\;& \text{amount of drift from buffer top} + \text{amount of drift from the interior of top half} \\ &+ \text{amount of drift from } s^* \end{aligned}$$

The first term is due to the effect of reflection at the buffer top, and is proportional to the fullness probability, $p_f(L)$; the second term is lower bounded by $E_{\pi_L}(\mu_{L,s}|L > s > \frac{L}{2})$. Since s^* contains the center state, the third term is lower bounded by a term proportional to $\pi_L(\frac{L}{2})$. Hence, if $p_f(L) = o(\frac{1}{L})$, then $-E_{\pi_L}(\mu_{L,s}|s > \frac{L}{2}) = \Omega(\frac{1}{L})$. By the symmetry of the control scheme, we get

$$E_\pi(|\mu_{L,s}|) = \Omega(\frac{1}{L}) \tag{3.1}$$

On the other hand, by a second-order Taylor series approximation of the distortion-rate function around $R = R_c$, the average normal distortion $D_q(L)$ is roughly given by $D_T(R_c) + \frac{D''_T(R_c)}{2} E_{\pi_L}(\mu^2_{L,s})$. (The first-order term disappears due to the symmetry in the control scheme.) Since $E_{\pi_L}(\mu^2_{L,s}) \geq (E_{\pi_L}(|\mu_{L,s}|))^2$, the desired lower bound follows from Eq. (3.1). □

We now give a very simple buffer control scheme which can achieve a near-optimal convergence rate for the normal distortion and at the same time having the fullness probability decay at a very fast rate. We first need the following lemma.

Lemma 3.4 *Let $(R_o, D_T(R_o))$ be a point in the interior of a single-quantizer achievable region of the distortion-rate curve. For every R in some neighborhood of R_o, assume that the optimal quantizer is unique and let Q_R be the optimal quantizer. Then*

$$\frac{dr^*(Q_R)}{dR}\,|_{R=R_o} = -\frac{2}{Var(|Q_{R_o}(Y)|)}$$

where, for $R \neq R_o$, $r^*(Q_R)$ is the unique nonzero root of the log moment generating function of the random variable $|Q_R(Y)| - R_o$.

Proposition 3.5 *Given any $K \geq 2$, there exists a buffer control scheme such that $p_f(L) = O(\frac{1}{L^K})$ and $D_q(L) - D_T(R_c) = \Theta(\frac{\ln^2 L}{L^2})$.*

Sketch of Proof. Consider the following buffer control scheme. For each buffer size L, use two quantizers Q_1^L and Q_2^L which achieve points on the distortion-rate curve at rates $R_c - \frac{C \ln L}{L}$ and $R_c + \frac{C \ln L}{L}$ respectively, with Q_1^L used in the top half of the buffer $(A_1)$and Q_2^L used in the bottom half (A_2). Note that in contrast to the scheme for the case when R_c is in the time-shared region, the quantizers are dependent on the buffer size L, with their rates approaching R_c as L becomes large. Using Wald's identity and renewal theory as in the proof of Prop. (3.2), one can show that the fullness probability is given by:

$$p_f(L) = O(\exp(-\frac{1}{2} r^*(Q_1^L) L))$$

where $r^*(Q_1^L)$ is the unique positive zero of the log moment generating function of the random variable $|Q_1^L(Y)| - R_c$. By Lemma (3.4), we have the approximation

$$r^*(Q_1^L) \approx \frac{2}{Var(|Q_{R_c}(Y)|)} \frac{C \ln L}{L}$$

By choosing $C = K \operatorname{Var}(|Q_{R_c}(Y)|)$, we get $p_f(L) = O(L^{-K})$.

When we approximate the average distortion of Q_1^L and Q_2^L via the Taylor series expansion of the distortion-rate function around $R = R_c$, the first-order term disappears due to the symmetry of the scheme, and we have the approximation $D_q(L) = D_T(R_c) + O(\frac{\ln^2 L}{L^2})$, thus giving us the claimed rate of convergence. □

The above result shows that we can almost achieve the lower bound of $1/L^2$ on the rate of convergence of the total distortion while keeping the fullness probability decaying at arbitrarily fast (polynomial) rate. Moreover, the buffer control scheme is exceedingly simple, using two quantizers only. We now compare this performance with the lower bound for the case when there is no buffer control.

Proposition 3.6 *Suppose we are only allowed to use one quantizer for a given buffer size. Then if the fullness probability* $p_f(L)$ *decays faster than* $1/L$*, the average distortion* $D_q(L)$ *converges to the optimal distortion no faster that* $1/L$*. Hence, for any single-quantizer scheme, the rate of convergence of the total distortion is lower bounded by* $1/L$*.*

Hence, by using two quantizers with opposing drifts, the rate of convergence is squared. Intuitively, the gain is because with one quantizer of negative drift, the buffer is empty a significant fraction of time, thus wasting the allocated channel rate. In the two quantizer case, this is avoided by keeping the buffer half-full most of the time. The quantizer for the lower half of the buffer actually operates at a rate greater than the channel rate, thus decreasing the average distortion.

4 Conclusions

We have considered the problem of buffer control of variable-rate quantization system for a fixed-rate channel. Asymptotically optimal control schemes , in the regime of large buffer sizes, have been derived for memoryless sources. Such optimal schemes have a simple form: they consist only of switching between two quantizers, one to be used when the buffer is more than half full and the other when the buffer is less than half full. An improvement over the best achievable without any buffer control is also demonstrated.

References

[1] T. Berger, M.U. Chang and S.Y. Tung-Kleinberg, "Quantization-permutation codes and buffer-adapted Huffman codes", in *Proc. 18th Allerton Conf. Comm. Contr. Comput.*, Monticello, IL, 1980, pp.433-436.

[2] N. Farvardin and J.W. Modestino, "Adaptive buffer-instrumented entropy-coded quantizer performance for memoryless sources", *IEEE Trans. Info. Theory*, vol. 32, no.1, Jan. 1986, pp. 9-22.

[3] R. Gallager and D. Bertsekas, *Data Networks*, 2nd Ed., Prentice Hall, 1992.

[4] D. Harrison and J.W. Modestino, "Analysis and further results on adaptive entropy-coded quantization", *IEEE Trans. Info. Theory*, vol. 36, no.5, Sept. 1990, pp. 1069-1088.

[5] P. Humblet, "Generalization of Huffman coding to minimize the probability of buffer overflow", *IEEE Trans. Info. Theory*, vol. 27, no. 2, Mar. 1981, pp. 230-232.

[6] F. Jelinek, "Buffer overflow in variable length coding of fixed rate sources", *IEEE Trans. Info. Theory*, vol.14, no.4, May 1968, pp. 490-501.

[7] A. Wald, "On cumulative sums of random variables", *Ann. Math. Statist.*, vol. 15, 1944, pp. 283-296.

[8] A. Wyner, "On the probability of buffer overflow under an arbitrary bounded input-output distribution", *SIAM J. Appl. Math.*, vol. 27, Dec. 1974, pp. 544-569.

Nonparametric Detection Applied to VQ-Based Speaker Verification

Michael A. Lund and C.C. Lee
Northwestern University
Department of Electrical Engineering and Computer Science
McCormick School of Engineering and Applied Science
Evanston, IL, 60208-3118, USA.
708-491-7375

Introduction

The problem of speaker verification can be formulated as a binary hypothesis testing problem, in which the input speech is attributed either to the claimed speaker or to the open class of all other speakers. Generally, the input speech is reduced to a set of parameters which is assumed to represent the speaker space, and this set is in turn reduced to a single decision statistic which is used in verifying the identity claim. A significant challenge in designing the final decision algorithm is to determine the threshold for acceptance vs. rejection. A problem arises due to the choice of inappropriate parametric models for the true speaker and imposter decision statistic distributions, because the speaker models (HMM's, DTW templates, or VQ code books), will necessarily exhibit a better match to the training data than to the data used in the test phase.

In contrast to parametric detection, which yields optimal performance for specific true-speaker and imposter distributions but may lead to severely degraded performance even for slight variations from the nominal distributions, nonparametric detectors are designed to deliver performance which is invariant over a large class of distributions. We propose a class of non-parametric detectors for use in speaker verification, and show for a text-dependent VQ verification system, that this class of detectors is superior to the parametric detector for the mean distortion.

In this paper, we present a nonparametric approach to speaker verification. The detector is based upon a generalization of the sign detector, and can be viewed as a test of location. In a slight misappropriation of radar terminology, we call this nonparametric detector a constant false acceptance rate (CFAR) detector, because it is designed such that a specified false acceptance rate will be achieved over a broad class of imposter distributions. The CFAR methodology allows us to set the decision threshold for our detector based solely on the imposter distribution, allowing us to sidestep the issue of unmodeled variations in the true-speaker characteristics. This strategy is effective to the extent that the imposter pool used in the training phase is large enough to model the characteristics of the entire class of potential imposters.

The Generalized Sign Test

This nonparametric test is based on the well-known sign detector for the median of a distribution [1]. The generalization allows us to focus on quantiles other than the median.

Consider d_{ij}, the distortion that results from encoding the i'th frame of the j'th keyword using an appropriate vector distortion measure. In the training phase, we determine the constant β_j to satisfy the relationship

$$p = \mathrm{Prob}\{d_{ij} < \beta_j\}, \tag{1}$$

based on the distribution of the d_{ij} for the imposter pool and a given value of p.

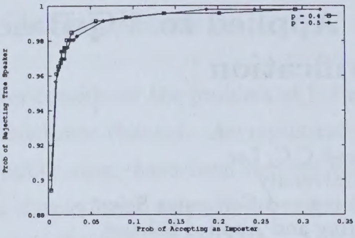


Figure 1: ROC for Generalized Sign Detector

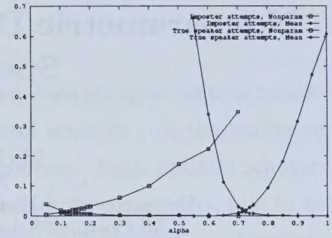


Figure 2: P{Error} for Generalized Sign Detector and Parametric Detector

In the test phase, we define the hypothesis pair

$$\begin{array}{lll} H_0: & \text{true speaker} & p > p_1 \\ H_1: & \text{imposter} & p = p_1, \end{array} \tag{2}$$

The sufficient statistic for discrimination between H_0 and H_1 is

$$t_j = |D|, \quad D = \{d_{ij} | d_{ij} > \beta_j\} \tag{3}$$

and the optimal α-level test of H_0 vs. H_1 is given as

$$t_j \begin{array}{lll} > & \tau & \text{choose } H_1 \\ \leq & \tau & \text{choose } H_0 \end{array} \tag{4}$$

where τ is the largest integer such that

$$\text{Prob}\{t_j \leq \tau\} \leq \alpha \tag{5}$$

where the above probability is calculated using the fact that t_j is a binomial random variable if the d_{ij} are assumed uncorrelated.

Verification Results

Results for a speaker verification system based on the concept of randomized phrase prompting [2] have been obtained on a database of 11 male speakers and can be summarized as follows:

Figure 1 contrasts the behavior of the generalized sign detector for $p_1 = 0.4$ and $p_1 = 0.5$. Notice that the discrimination power in the $p_1 = 0.4$ case is greater than for the median detector because distortion values just below the median will still favor the imposter hypothesis, while extremely small values will favor the true speaker hypothesis.

The performance of the generalized sign test is compared with the performance of the parametric detector in Figure 2, where the mean of the true speaker distribution is assumed to be $(1 + \alpha)\mu_0$, where μ_0 is determined from the training data. Notice that the generalized sign detector exhibits much more robustness against variations from the optimal value of α than the optimal parametric detector does.

References

[1] H. V. Poor. *An Introduction to Signal Detection and Estimation.* Springer Verlag, New York, 1988.

[2] A. Higgins, L. Bahler, and J. Porter. Speaker verification using randomized phrase prompting. *Digital Signal Processing*, 1:89–106, 1991.

EFFECT OF CHANGE IN NOISE DISTRIBUTION ON DISTRIBUTED HYPOTHESIS TESTING SYSTEMS

Shivanand I. Akkihal
Department of Electrical and Computer Engineering
The University of Texas at Austin, Austin Texas 78712
and
Gary L. Wise
Department of Electrical and Computer Engineering
and
Department of Mathematics
The University of Texas at Austin, Austin Texas 78712

I. Abstract

A distributed hypothesis testing system for a constant known signal with N identical sensors and mutually independent and identically distributed noise distributions is considered. The system is designed under the assumption of a certain noise distribution, and then used for some other noise distribution. The behavior of the overall false alarm probability and the overall detection probability is studied under this type of change in the noise distribution. It is shown that for a fixed sensor false alarm probability α and fixed sensor detection probability β, the behavior of the system performance is controlled by the value of the noise distribution function at two points. Plots of this behavior will be given for a system designed under the assumption of Gaussian noise and used for Laplace noise.

II. Introduction

A considerable amount of research has been done on the problem of hypothesis testing. This research was concentrated on two main topics. One of the topics was hypothesis testing with a single sensor. The other topic was hypothesis testing with more than one sensor, where the sensors take the observations (i.e. the random variables of concern) and transmit them to a central processor. Hypothesis testing with a single sensor did not yield sufficiently high performance. The drawback of the multiple sensor hypothesis testing was the large number of bits required to transmit the sensor observations to the central processor. Distributed hypothesis testing emerged as a solution for the drawbacks of both the above mentioned methods of hypothesis testing. Since then it has received a considerable amount of attention. The reason for this was improved performance, less number of bits required to transmit data from the sensors to the fusion center, and in practice it was better than the multiple sensor hypothesis testing system. The problem of distributed hypothesis testing is considered in [1]–[6] under different assumptions. Tenny and Sandell [1] considered the problem for the Bayesian method. Sadjadi [2] extended the problem for m hypotheses instead of simple hypothesis testing.

Later Thomopoulos et al. [6] considered the problem for a Neyman-Pearson test at the sensors and the fusion center. In this paper we will consider the problem of using a distributed hypothesis testing system under a certain noise distribution when designed for some other noise distribution.

We consider the simple hypothesis testing problem for a constant known signal with a distributed hypothesis testing system with N sensors. Note that even though this is a hypothesis testing problem (see [9]) we will be using the conventional notation from detection theory for simplicity and consistency with the existing literature. For simplicity we assume hypothesis H_1 denotes the hypothesis corresponding to signal present, and hypothesis H_0 denotes the hypothesis corresponding to signal absent. Based on an observation in the noisy environment we need to accept one of these two hypotheses. Let u_i denote the decision of the i^{th} sensor based on the observation it takes at that time. We assume that each sensor transmits a one bit decision to the fusion center. The i^{th} sensor transmits $u_i = 1$ if it favors H_1 otherwise it transmits $u_i = 0$. Let $\mathbf{u} = (u_1, u_2, \ldots, u_N)$ denote the vector of the field sensor decisions. The fusion center makes a final decision based on the one bit decisions of all the sensors. We assume that the sensors and the fusion center employ a Neyman-Pearson(NP) test. We denote the distribution of $\mathbf{u}$ under hypothesis H_0 by P_0 and the distribution of $\mathbf{u}$ under hypothesis H_1 by P_1. The likelihood ratio at the fusion center is given by a ratio of Radon-Nikodym derivatives

$$\Lambda(\mathbf{u}) = \frac{p_1(u_1, \ldots, u_N)}{p_0(u_1, \ldots, u_N)} \tag{1}$$

where $p_i = dP_i/d\lambda$ and where λ is a σ-finite dominating measure on the Borel subsets of $\mathbf{R}^N$ (which always exists, e.g. $\lambda = P_0 + P_1$). The final decision u_f is given by the NP test at the fusion center by

$$u_f = \begin{cases} 1 & \text{if } \Lambda(\mathbf{u}) > t_f \\ 0 & \text{if } \Lambda(\mathbf{u}) < t_f \\ 1 & \text{with probability } \omega \text{ if } \Lambda(\mathbf{u}) = t_f \end{cases} \tag{2}$$

We further assume that the sensors make independent decisions on the observations they take. The noise is assumed to be mutually independent and identically distributed at each sensor. Due to the mutual independence, the likelihood ratio at the fusion center factors and is given by

$$\Lambda(\mathbf{u}) = \prod_{i=1}^{N} \frac{q_1(u_i)}{q_0(u_i)}, \tag{3}$$

where the $q_j(u_i)$'s are Radon-Nikodym derivatives as before. Using the above likelihood ratio the fusion center makes a decision favoring H_0 or H_1 according to the NP test.

We assume that all the sensors are operating at the same false alarm probability α and the same detection probability β. And we also assume that the overall false alarm probability $P_F = \alpha$, the common sensor false alarm probability. As noted in [6] under either hypothesis it is easier to compute the distribution of $\log \Lambda(\mathbf{u})$ than to compute the distribution of $\Lambda(\mathbf{u})$. The likelihood ratio at the fusion center has a binomial distribution with parameters N and α under H_0, and parameters N and β, under H_1. Note that since we have assumed that all sensors are identical, the likelihood ratio at the fusion center takes $N+1$ different values. The values the likelihood ratio takes at the fusion center are $\frac{\beta^i(1-\beta)^{N-i}}{\alpha^i(1-\alpha)^{N-i}}$ for $i = 0, 1, \ldots, N$. The LR takes these values with probability $\binom{N}{i}\beta^i(1-\beta)^{N-i}$ under H_1 and with probability $\binom{N}{i}\alpha^i(1-\alpha)^{N-i}$ under H_0. The threshold at the fusion center is chosen so that $P_F = \alpha$, where α is a predetermined value. Denoting the randomization factor by ω, the expressions for the overall false alarm probability P_F and the overall detection probability P_D are given by

$$P_F = \omega \binom{N}{k} \alpha^k(1-\alpha)^{N-k} + \sum_{i=k+1}^{N} \binom{N}{i} \alpha^i(1-\alpha)^{N-i} \tag{4}$$

and

$$P_D = \omega \binom{N}{k} \beta^k(1-\beta)^{N-k} + \sum_{i=k+1}^{N} \binom{N}{i} \beta^i(1-\beta)^{N-i} \tag{5}$$

where, $0 \leq k \leq N$ and $0 \leq \omega \leq 1$.

The threshold t_f at the fusion center is chosen in such a way that the overall false alarm probability P_F is equal to the predetermined value. Note that this value of t_f is computed assuming a certain noise distribution. For easier understanding assume the system was designed under the assumption of Gaussian noise. Then the fusion threshold t_f is computed according to the expression involving the Gaussian noise. If m is the signal strength and the noise is iid Guassian with zero mean and unit varaince then the observation has the distribution $N(0,1)$ under H_0, and the distribution $N(m,1)$ under H_1, where $N(m,\sigma^2)$ denotes a Gaussian distribution with mean m and variance σ^2. From the expression for the overall false alarm probability P_F we get the expression for ω as

$$\omega = \frac{P_F - \sum_{i=k+1}^{N} \binom{N}{i} \alpha^i(1-\alpha)^{N-i}}{\binom{N}{k} \alpha^k(1-\alpha)^{N-k}} \tag{6}$$

where α and β are computed assuming the noise distribution is standard Gaussian.

III. Effect due to change in the noise distribution

We see from (4)-(6) that the value of the randomization factor ω, the overall false alarm probability and the overall detection probability depend on the values of the sensor false alarm probability α and the sensor detection probability β. But we know that α is equal to one minus the value of the distribution function under hypothesis H_0 at the sensor threshold t. Similarly the value of the sensor detection probability β is equal to one minus the value of the distribution function under hypothesis H_1 at the sensor threshold t, which is equal to the value of the distribution under hypothesis H_0 at $t-m$. Thus the value of the α and β depend on the value of the distribution function under hypothesis H_0 at t and $t-m$. Thus in case of two different noise distributions, the values of α and β remain unchanged if the two noise distribution functions have same values at t and $t-m$. Thus it follows that if we know the operating range of α and β then we know the subset of the real line over which the two noise distributions have to agree for there to be no change in the values of α and β. Conversely if we have two radically different noise distributions but which agree on the subset of the real line of interest then there will be no difference in the α and β values obtained with those two noise distributions.

For definiteness let us assume the distributed hypothesis testing system is designed under the assumption of Gaussian noise distribution with zero mean and unit variance. Let this system be used under some noise distribution other than the Gaussian distribution. For simplicity assume it is used under the Laplace distribution having zero mean and unit variance. In practice we may encounter this type of situation when we use a system designed for a certain environmental condition but used for some other environmental condition. The actual noise distribution may be different than the design noise distribution due to change in the physical parameters in nature. Let α_l and β_l denote the new sensor false alarm probability and new sensor detection probability under Laplace noise. The values of α_l and β_l will in general be different than the values of α and β respectively since the values of their distribution functions are not same at t and $t - m$ in general. The observations will now have distribution $L(0,1)$ under H_0 and distribution $L(m,1)$ under H_1 where $L(a,b)$ denotes Laplace distribution with mean a and variance b. Due to these new values of the sensor false alarm probability and the sensor detection probability, the LR at the fusion center will have the binomial distribution with parameters N and α_l under H_0 and with parameters N and β_l under H_1. Since α_l and β_l have in general different values than α and β, the values the LR takes at the fusion center are different than the corresponding values under the Gaussian noise distribution.

While designing the system the sensor threshold and the fusion threshold are determined according to the assumed noise distribution. At the fusion center, due to the discrete distribution of the LR, we have randomization of the NP test for all but $N + 1$ different values of P_F. This means the fusion center threshold t_f takes one of the values the LR takes at the fusion center. Now using this system under some other noise distribution will result in different LR values at the fusion center as explained above. The fusion center threshold t_f is the same as the design fusion threshold. Due to a change in the noise distribution the LR values taken at the fusion center are different then their corresponding values under the assumed design noise distribution. Due to this change, the new overall false alarm probability and the new overall detection probability do not have a contribution by the randomization factor. Thus there is a significant difference in the overall false alarm probability and overall detection probability compared to the design values.

Let us denote the LR value under the design noise which is equal to the fusion threshold value by t_g. Let us denote the corresponding LR value under a different noise distribution by t_l. Due to the binomial distribution, the value of LR at the fusion center $t_g = \binom{N}{k} \frac{\beta^k(1-\beta)^{N-k}}{\alpha^k(1-\alpha)^{N-k}}$ for some k such that $0 \leq k \leq N$. When using this system under some other noise distribution the corresponding LR value t_l at the fusion center is given by $t_l = \binom{N}{k} \frac{\beta_l^k(1-\beta_l)^{N-k}}{\alpha_l^k(1-\alpha_l)^{N-k}}$. Depending on the values of α and α_l, and, β and β_l we get three cases. The first one is when $t_g = t_l$. In this case there is no significant change in the overall false alarm probability and the overall detection probability as long as α and α_l, and β and β_l are close to each other respectively. The second case is when $t_l < t_g$. In this case, since the fusion threshold is equal to t_g, the overall false alarm probability and the overall detection probability will lose their corresponding contribution by randomization. In this case if the value of ω is close to one then the amount lost is close to $\binom{N}{k}\alpha_l^k(1-\alpha_l)^{N-k}$ for the overall false alarm probability and close to $\binom{N}{k}\beta_l^k(1-\beta_l)^{N-k}$ for the overall detection probability. The third case is when $t_l > t_g$. In this case the effective randomization factor is changed to one. If the randomization factor is close to zero then the new overall false alarm probability is increased approximately by $\binom{N}{k}\alpha_l^k(1-\alpha_l)^{N-k}$, and the overall detection probability is increased approximately by $\binom{N}{k}\beta_l^k(1-\beta_l)^{N-k}$ In some cases this increase in the overall false alarm probability may be significantly high as will be shown in the example.

Example

Assume a distributed hypothesis testing system with $N = 5$ sensors is designed. Let the design noise distribution be the zero mean and unit variance Gaussian distribution. Let the signal strength be $m = 3$. The sensor threshold t is computed based on the set overall false alarm probability $P_F = \alpha$. It is given by $t = Q^{-1}(\alpha)$, where $Q(t) = \frac{1}{\sqrt{2\pi}}\int_t^\infty e^{-x^2/2}\,dx$. The sensor detection probability β is obtained by $\beta = Q(t - m)$. The fusion center threshold is obtained by (4) such that the overall false alarm probability is equal to some set value. Let us denote the fusion threshold by t_g. The value of t_g is equal to one of the LR values at the fusion threshold off a finite set.

Now suppose this system is used under the Laplace noise distribution with zero mean and unit variance. The sensors use the design threshold t for computing the values of the new sensor false alarm probability α_l and the new detection probability β_l. Thus the actual LR values at the fusion center are different then their corresponding values in the case of the Gaussian noise distribution. Depending on the shift in the LR values with respect to the corresponding design LR values the overall false alarm probability and the overall detection probability are changed.

Figure 1 shows the LR values and the probability they take those values. The solid line denotes the design values and the dashed line denotes the values when used for the Laplace noise distribution. Figure 2 show the plot of the randomization factor ω against α and the plot of direction of shift in the LR value under Laplace noise corresponding to the LR value under Gaussian noise which is equal to the fusion threshold. The value $sign = -1$ indicates that the shift is negative, that is the LR value under Laplace noise (corresponding to t_g in Gaussian noise case) is less than the corresponding LR value under Gaussian noise, and $sign = 1$ indicates the shift is positive. Figure 3 shows the plots of the overall false alarm probability P_F against α. Figure 4 shows the plots of the overall detection probability P_D against α. In Figure 3 and Figure 4 the plot G-G denotes the plot for a system designed and used under Gaussian noise, the plot L-L denotes the plot for a system designed and used under Laplace noise, and the plot G-L denotes a system designed for Gaussian noise but used for Laplace noise. Figure 3 shows graphically the sensitivity of the overall false alarm probability P_F due to change in noise distribution. There is significant increase in the value of the overall false alarm probability P_F in a certain range of α when the system is designed under Gaussian noise and used under Laplace noise. The new overall false alarm probability has a large jump in its value at around $\alpha = 0.087$. This is the value of α where the distribution functions of the Gaussian noise and the Laplace noise agree at points t and $t - m$ for this example. When operating at this α value the change in the noise distribution function from unit normal distribution to Laplace distribution with zero mean and unit variance does not change the performance of the system. This also is the value of α where the sign of the shift in the LR value changes from positive to negative, as shown in Figure 2. This jump in the value of P_F remains until $\alpha = 0.131$. As shown by Figure 2 this is the α value (0.131) where the fusion threshold changes from one value of the LR to another value of LR. Thus there exist intervals of α values in which the change in noise distribution will result in a significantly large value for P_F. The effect on overall detection probability P_D is not as significant as on P_F because of the very small value of probability of the LR value which is equal to the fusion threshold under hypothesis H_1. Thus in operating a distributed hypothesis testing system in the above mentioned situation care should be taken that the operating α value does not lie in an interval with significant deterioration in performance due to a change in the noise distribution.

IV. Conclusion

The problem of change in the noise distribution in a distributed hypothesis testing system is considered. The system is assumed to be designed under some noise distribution and used under some other noise distribution. The behavior of P_F and P_D under such a change is studied. It is shown that there exist intervals of operating α values in which the deterioration in the P_F value is significant due to a change in the noise distribution. The change in P_F and the P_D values is due to the change in the effective randomization factor at the fusion center. It is shown that given a compact subset of $(0, 1)$ as the operating range of α there exists a bounded subset of the real line, such that the behavior of the noise distribution functions in this subset completely governs the behavior of the system when designed for one of the noise distributions and used for the other. An example is presented in which the system is designed under Gaussian noise and used under Laplace noise, when the number of sensors is equal to 5. It is shown that when operating in the interval $(0.089, 0.131)$ the deterioration in the P_F values is significant. These results can be extended in a straightforward manner to cases of different noise distributions. We conclude that before using a distributed hypothesis testing system under some noise distribution other than the design noise distribution care should be taken so that the operating α value does not lie in the interval with significant deterioration in the P_F value. A moment of thought will reveal that this sensitivity of the performance is due to the discrete nature of the likelihood ratio at the fusion center, which is binomially distributed.

Acknowledgement

The authors wish to greatfully acknowledge the support of the Jewel McAlister Smith Professorship in Engineering at The University of Texas at Austin.

V. References

[1] Tenny, R. R., and Sandell, N. R., Jr. Detection with distributed sensors. *IEEE Transactions on Aerospace and Electronic Systems*, AES-17, July 1981, pp. 501–510.

[2] Sadjadi, F. A. Hypotheses testing in distributed environment. *IEEE Transactions on Aerospace and Electronic Systems*, AES-22, No. 2, Mar. 1986, pp. 134–137.

[3] Chair, Z., and Varshney, P. K. Optimal data fusion in multiple sensor detection systems. *IEEE Transactions on Aerospace and Electronic Systems*, AES-22, Jan. 1986, pp. 98–101.

[4] Srinivasan, R. Distributed radar detection theory. *IEE proceedings*, 133, Pt. F, Feb. 1986, pp. 55–60.

[5] Reibman, A. R., and Nolte, L. W. Optimal detection and performance of distributed sensor systems. *IEEE Transactions on Aerospace and Electronic Systems*, AES-23, No. 1, Jan. 1987, pp. 24–30.

[6] Thomopoulos, S. C. A., Viswanathan, R., and Bougoulias, D. C. Optimal decision fusion in multiple sensor systems. *IEEE Transactions on Aerospace and Electronic Systems*, AES-23, Sept. 1987, pp. 644–653.

[7] Lee, C. C., and Chao, J. J. Optimal local decision space partioning for distributed detection. *IEEE Transactions on Aerospace and Electronic Systems*, vol. 25, No. 4, July 1989, pp. 536–544.

[8] Thomopoulos, S. C. A., Viswanathan, R., and Bougoulias, D. C. Optimal distributed decision fusion. *IEEE Transactions on Aerospace and Electronic Systems*, vol. 25, No. 5, Sept. 1989, pp. 761–765.

[9] Wise, G. L., Hall, E. B. A common misconception in detection theory. *Proceedings of the 1993 Conference on Information Sciences and Systems*, Mar. 24–26, 1993.

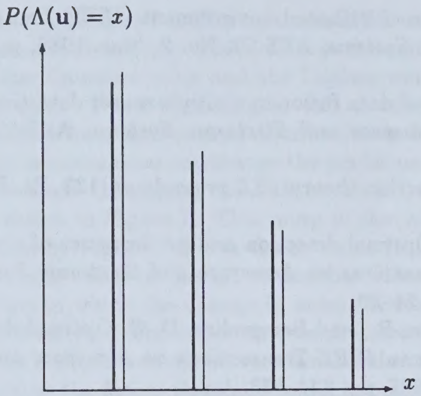


Figure 1. Probability of the LR values for $N = 3$ for two different noise distributions. The thin lines are for the Gaussian noise and the thick lines are for the Laplace noise.

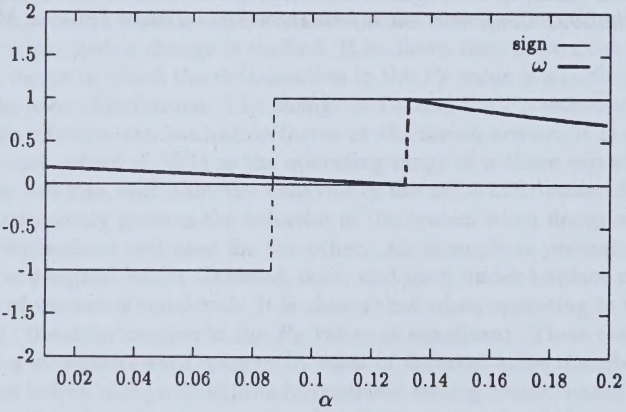


Figure 2. Plots of ω and sign of shift in LR value against α.

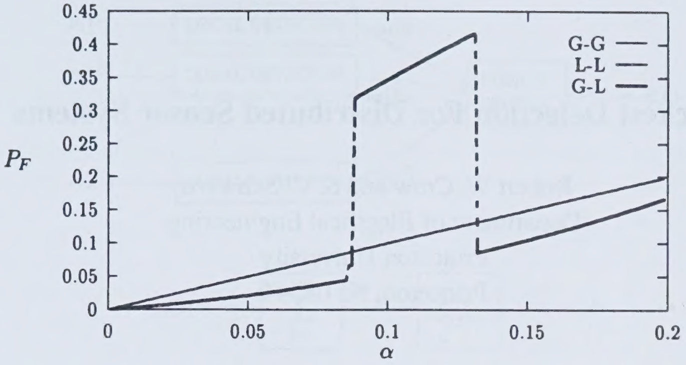


Figure 3. Plots of P_F against α.
G-G : System designed and used for Gaussian noise.
L-L : System designed and used for Laplace noise.
G-L : Syetm designed for Gaussian noise, but used for Laplace noise.

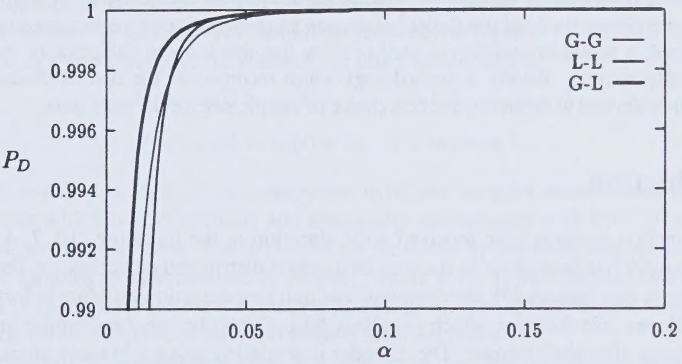


Figure 4. Plots of P_D against α.
G-G : System designed and used for Gaussian noise.
L-L : System designed and used for Laplace noise.
G-L : Syetm designed for Gaussian noise, but used for Laplace noise.

Quickest Detection For Distributed Sensor Systems [*]

Robert W. Crow and S. C. Schwartz
Department of Electrical Engineering
Princeton University
Princeton, NJ 08544

Abstract

The quickest distributed detection problem for an unknown disorder time and local decisions produced by a simple summer-comparator structure is examined. The maximum likelihood (ML) optimal procedure is derived, along with a suboptimal procedure which requires less computation to implement. A third procedure which is suitable for the case when the magnitude of the disorder is unknown is also examined. The performance of the procedures is computed by utilizing the Markov property of the test statistics. A simple method for choosing the local thresholds based upon an asymptotic performance measure is developed; a sensitivity analysis is used to show that this leads to performance that is practically optimal. Finally, a methodology which incorporates the cost of channel bandwidth is devised to determine the best choice of sample size for the local tests.

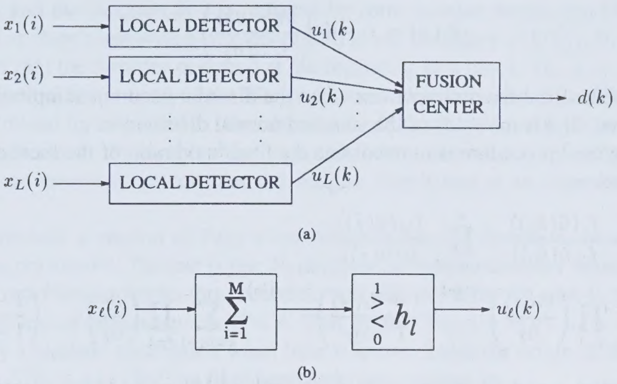
1 Introduction

Distributed detection systems have received wide attention in the literature [10, 7, 4, 3, 11]. However, little work has been done in the area of *quickest* distributed detection. In Teneketzis [8] and Teneketzis and Varaiya [9], the decentralized quickest detection problem is formulated by defining a Bayes cost function which penalizes false alarms before the disorder and large delays in detection after the disorder. The disorder is modelled using a Markov chain, where the conditional probability of the jump occurring at time $i + 1$ given that it did not occur at time i is some fixed value, presumably known or inferred from previous data. It is shown in [9] that, for the case where the cost function is not separable with respect to the local decisions, the local thresholds are the solutions to a set of coupled dynamic programming equations and are time-varying. A separable cost function is also considered; this results in fixed local thresholds, although with a longer delay in detection.

In this paper, we examine several fusion rules for the case where the local detectors simply consist of some type of integrator structure followed by a comparator with a fixed threshold. We assume the disorder time to be unknown, leading to maximum likelihood rather than Bayesian procedures. This assumption is usually made in radar problems, or in other problems where the disorder occurs only after a long period of time. Several of the fusion procedures we consider assume knowledge of the signal strengths before and after the disorder; however, we also examine some procedures which are applicable when this information is not available.

---

[*] Research supported in part by the Office of Naval Research under contract N00014-88-K-0162 and by the National Science Foundation under contract CDA-91-21709



(a)

(b)

Figure 1: (a) Block diagram of distributed system. (b) Structure of the local detectors.

2 Problem statement

The decentralized detection system under consideration is shown in Figure 1. Here $\{x_\ell(i)\}_{i=1}^n$ is the sequence of samples received by sensor ℓ up to time n, where $\ell = 1, 2, \ldots, L$, and L is the total number of sensors. The sampling frequency is fixed at $f_s = 1/T_s$. The disorder is modelled as a step change in the mean of the observables; that is:

$$\begin{aligned} H_0: \quad & x_\ell(i) = n_\ell(i), && i = 1, 2, \ldots, m-1 \\ H_1: \quad & x_\ell(i) = n_\ell(i) + s_\ell, && i = m, m+1, \ldots \end{aligned}$$

for each sensor $\ell = 1, 2, \ldots, L$, where the $n_\ell(i)$ are samples from a zero mean Gaussian distribution which is both spatially and temporally uncorrelated with $\mathrm{E}[n_\ell^2(i)] = \sigma_\ell^2$, and m is the unknown *disorder time*. The decisions are produced by the local detectors at a rate of one every M samples and are denoted by $\{u_\ell(k)\}$, where $u_\ell(k)$ is the indicator that the sum of the samples in block k for sensor ℓ exceeds a threshold h_ℓ. The global decisions are produced at the fusion center based upon past and present local decisions.

The goal of the overall procedure is to minimize the expected time to detect the disorder after its occurrence, D, subject to a lower bound on the mean time between false alarms before the disorder, T. One can also define the *asymptotic performance measure*

$$\eta = \lim_{T \to \infty} \frac{\log T}{D}$$

which will be shown to be useful in determining the thresholds $\{h_\ell\}$ of the local tests.

3 Derivation of the fusion rules

It is assumed that the post-disorder signals $\{s_\ell\}$ are known. Let $f_\mu(u_1, u_1, \ldots, u_L), \mu \in \{0, 1, \ldots, M\}$ denote the distribution of the local decisions $\bar{u} = \{u_\ell\}_{\ell=1}^L$ for a block where the first $M - \mu$ samples of the block are drawn from H_0 and the last μ are from H_1. Thus, the decisions are distributed as f_0 for blocks $1, 2, \ldots, k_0 - 1$ and as f_M for blocks $k_0 + 1, k_0 + 2, \ldots$, while at the *change block* k_0 the distribution is f_μ for some unknown $\mu \in \{1, 2, \ldots, M\}$. The decisions resulting from block k have distribution $\Pr\{u_\ell(k) = 1\} = \theta_\ell(\mu)$, where

$$\theta_\ell(\mu) = 1 - \Phi\left(\frac{h_\ell - \mu s_\ell}{\sqrt{M}\sigma_\ell}\right)$$

is the power of the local fixed sample test when the disorder occurs μ samples before the end of the block, and $\Phi(\cdot)$ is the c.d.f. of the standard normal distribution.

The ML optimal procedure is to maximize the likelihood ratio of the local decisions up to the current block n,

$$\begin{aligned}\mathcal{L}_n(\mu, k_0) &= \frac{f_\mu(\bar{u}(k_0))}{f_0(\bar{u}(k_0))} \cdot \prod_{j=k_0+1}^{n} \frac{f_M(\bar{u}(j))}{f_0(\bar{u}(j))} \\ &= \prod_{\ell=1}^{L}\left(\frac{\theta_\ell(\mu)}{\alpha_\ell}\right)^{u_\ell(k_0)}\left(\frac{1-\theta_\ell(\mu)}{1-\alpha_\ell}\right)^{1-u_\ell(k_0)} + \sum_{k=k_0+1}^{n}\prod_{\ell=1}^{L}\left(\frac{\beta_\ell}{\alpha_\ell}\right)^{u_\ell(k)}\left(\frac{1-\beta_\ell}{1-\alpha_\ell}\right)^{1-u_\ell(k)}\end{aligned}$$

over the quantities μ and k_0, where $\alpha_\ell = \theta_\ell(0)$ and $\beta_\ell(M) = \theta_\ell(M)$; a disorder is declared if this quantity exceeds a fixed threshold. This is also called the generalized likelihood ratio test (GLRT). A recursive implementation of this test can be derived as follows. Define the test statistic

$$S_n = \max_{1\leq k_0 \leq n}\ \max_{1\leq \mu \leq M}\ \log \mathcal{L}_n(\mu, k_0)$$

The GLRT is then to declare a disorder at block n in case $S_n > h$, where the threshold h of the test is chosen to satisfy a false alarm condition. We define

$$\phi(\bar{u}(k)) = \max_{1\leq\mu\leq M}\sum_{\ell=1}^{L}\left\{u_\ell(k)\log\left(\frac{\theta_\ell(\mu)}{\alpha_\ell}\right) + (1-u_\ell(k))\log\left(\frac{1-\theta_\ell(\mu)}{1-\alpha_\ell}\right)\right\}$$

for each block k, so that

$$S_n = \max_{1\leq k_0\leq n}\ \phi(\bar{u}(k_0)) + (n-k_0)d + \sum_{k=k_0+1}^{n}\sum_{\ell=1}^{L} c_\ell u_\ell(k)$$

where $c_\ell = \log\left(\frac{\beta_\ell(1-\alpha_\ell)}{\alpha_\ell(1-\beta_\ell)}\right)$ and $d = \sum_{\ell=1}^{L}\log\left(\frac{1-\beta_\ell}{1-\alpha_\ell}\right)$. Finally, one can observe that S_{n+1} can be expressed in terms of S_n and the new data $\bar{u}(n+1)$. The recursive test is then

$$\mathcal{P}_1 : \left\{\begin{array}{lcl} S_n &=& \max\left\{S_{n-1} + g_1(\bar{u}(n)), \phi(\bar{u}(n))\right\}, \quad S_0 = 0 \\ \tilde{N}_1 &=& \inf\{n \mid S_n > h\}\end{array}\right.$$

where

$$g_1(\bar{u}(n)) \triangleq \sum_{\ell=1}^{L} c_\ell u_\ell(n) + d$$

Here, $\tilde{N}_1$ is the stopping time of the test. Note that the stopping time is expressed in blocks (indicated by the tilde); the conversion to samples is done using the following expressions:

$$\begin{aligned}\mathrm{E}_0 N &= M \cdot \mathrm{E}_0 \tilde{N} \\ \mathrm{E}_1 N &= \mu + \left(\mathrm{E}_1 \tilde{N} - 1\right) M\end{aligned}$$

One can notice the similarity of this test to Page's procedure:

$$\mathcal{P}_2 : \left\{\begin{array}{lcl} S_n &=& \max\left\{S_{n-1} + g_2(\bar{u}(n)), 0\right\}, \quad S_0 = 0 \\ \tilde{N}_2 &=& \inf\{n \mid S_n > h\}\end{array}\right.$$

where $g_2 = g_1$ and the function $\phi(\cdot)$ is replaced by zero; in other words, the ML optimal test can be viewed as Page's test with a data dependent lower boundary $\phi(\bar{u}(k))$. In fact, if it were known *a priori* that the disorder occurred at the beginning of a block, i.e. $\mu = M$, Page's test would be the optimal test not only in the ML sense, but also in the sense of Lorden [5]; that is, D would be minimized for any fixed T. One drawback of the optimal test is the need to compute $\phi(\bar{u}(k))$ for each block. Since this requires a maximization over a potentially large number of points, we are motivated to consider the simpler Page's test as an alternative suboptimal procedure.

We now consider a version of Page's test which is suitable for the case where the jump magnitudes are not known. The test is just $\mathcal{P}_2$ designed to detect a disorder when $s_\ell = \delta_\ell$; since the likelihood ratio is monotonic, this procedure will also work for the case $s_\ell > \delta_\ell$. First, we assume without loss of generality that $\delta_\ell \equiv \delta$. This implies $\theta_\ell(\mu) = \theta(\mu)$, $\alpha_\ell = \alpha$, and $\beta_\ell = \beta$. This is actually a realistic assumption when little is known about the origin of the disturbance. Define $v(k) = \sum_{\ell=1}^{L} u_\ell(k)$. The log likelihood ratio now reduces to

$$\log \mathcal{L}_n = v(k_0) \log\left(\frac{\theta(\mu)}{\alpha}\right) + (1 - v(k_0)) \log\left(\frac{1-\theta(\mu)}{1-\alpha}\right) + (n - k_0)d + c \sum_{k=k_0+1}^{n} v(k)$$

where $c \stackrel{\Delta}{=} \log\left[\frac{\beta(1-\alpha)}{\alpha(1-\beta)}\right]$ and $d \stackrel{\Delta}{=} L \log\left(\frac{1-\beta}{1-\alpha}\right)$. The version of Page's procedure detecting changes of unknown signal strength is then:

$$\mathcal{P}_3 : \quad \left\{ \begin{array}{lcl} S_n & = & \max\{S_{n-1} + g_3(v(n)), 0\}, \quad S_0 = 0 \\ \tilde{N}_3 & = & \inf\{n \mid S_n \geq h\} \end{array} \right.$$

where $g_3(v(n)) = cv(n) + d$. This is referred to as Hinkley's test in the literature.

Finally, we consider the case where all of the data is available at the fusion center, which we call the *non-distributed* case. The optimal non-distributed procedure in the sense of Lorden [5] is

$$\mathcal{P}_0 : \quad \left\{ \begin{array}{lcl} S_n & = & \max\{S_{n-1} + z_n, 0\}, \quad S_0 = 0 \\ N_0 & = & \inf\{n \mid S_n > h\} \end{array} \right.$$

where

$$z_j \stackrel{\Delta}{=} \sum_{\ell=1}^{L} \sigma_\ell^{-2} \left\{ x_\ell(j) s_\ell - \frac{1}{2} s_\ell^2 \right\}$$

It will be shown that, while this test requires the largest channel bandwidth, it also yields the best performance because the information is not reduced locally. It is therefore included as a benchmark to which the other procedures are compared.

The performance of procedures $\mathcal{P}_0$ - $\mathcal{P}_3$ can be computed by utilizing the fact that the test statistic for each has the Markov property; our approach is similar to that in [1]. An expression for the probability that the procedure at stage n will reach stage $n+1$ is derived and expressed in terms of the probability transition matrix. The expected stopping time is then just the sum of these probabilities. The details of this computation can be found in [2].

4 Choosing the local thresholds

Let $\bar{h}_{loc} = [h_1, h_2, \ldots, h_L]$ denote the vector of fixed local thresholds. In Section 1, we defined the asymptotic performance measure, η. It is shown in [2] that η is the same for both the ML optimal and Page procedures, and that a lower bound $\tilde{\eta}$ on η can be derived, where

$$\tilde{\eta} = \frac{1}{M}\,\mathrm{E}_1[g(\bar{u})] \tag{1}$$

where $g = g_i$ as defined for $\mathcal{P}_i$, $i = 1, 2, 3$.

We would like to choose the local thresholds so as to maximize this lower bound on asymptotic performance; this amounts to maximizing $\mathrm{E}_1[g(\bar{\nu})]$ since M is a constant. We have

$$\begin{aligned}
\max_{\bar{h}_{loc}} \mathrm{E}_1[g(\bar{\nu})] &= \max_{\bar{h}_{loc}} \mathrm{E}_1\left[\sum_{\ell=1}^{L}\left\{\nu_\ell \log\left(\frac{\beta_\ell}{\alpha_\ell}\right) + (1-\nu_\ell)\log\left(\frac{1-\beta_\ell}{1-\alpha_\ell}\right)\right\}\right] \\
&= \max_{\bar{h}_{loc}} \sum_{\ell=1}^{L} \mathrm{E}_1\left\{\nu_\ell \log\left(\frac{\beta_\ell}{\alpha_\ell}\right) + (1-\nu_\ell)\log\left(\frac{1-\beta_\ell}{1-\alpha_\ell}\right)\right\} \\
&= \max_{\bar{h}_{loc}} \sum_{\ell=1}^{L} D_b(\beta_\ell, \alpha_\ell)
\end{aligned}$$

where $\mathrm{E}_1[\nu_\ell] = \beta_\ell$ and $D_b(a, b) \triangleq a\log(\frac{a}{b}) + (1-a)\log(\frac{1-a}{1-b})$ is just the binary discrimination function from information theory. Finally, the convexity of D_b in each variable allows us to break up the maximization termwise as:

$$\max_{\bar{h}_{loc}} \mathrm{E}_1[g(\bar{\nu})] = \sum_{\ell=1}^{L} \max_{h_\ell} D_b(\beta_\ell, \alpha_\ell)$$

Thus, the problem of globally maximizing the asymptotic lower bound amounts to selecting the local thresholds to maximize the binary discrimination for each sensor.

The local threshold h_ℓ is related to α_ℓ and β_ℓ for $\ell = 1, 2, \ldots, L$ via the equations:

$$\alpha_\ell = 1 - \Phi\left(\frac{h_\ell}{\sqrt{M}\sigma_\ell}\right) \quad \text{and} \quad \beta_\ell = 1 - \Phi\left(\frac{h_\ell - M s_\ell}{\sqrt{M}\sigma_\ell}\right)$$

The function $D_b(\beta_\ell(h_\ell), \alpha_\ell(h_\ell))$ has a unique maximum over $[0, \infty]$, so the optimal threshold is easy to compute.

5 Example of performance computation

Here, we illustrate the performance of procedures $\mathcal{P}_0$-$\mathcal{P}_3$ for a system with $L = 4$ sensors, where the noise at each sensor is standard Gaussian. We will consider the case where the signal to noise ratios at each sensor after the disorder are $0, -3, -6$, and -10 dB (note: the SNR at sensor ℓ is $10\log(s_\ell^2)$ for this example).

The average and worst-case performance of the ML optimal, Page, and non-distributed procedures are compared for the strong signal case in Figure 2. The performance of the non-distributed procedure is uniformly better than that of the other procedures, as expected. Note that both the average and worst-case performances of the ML optimal and Page procedures are virtually identical, with the ML optimal winning out only by a small amount in some places. We also see that the performance degrades with increasing blocksize, and the best choice is $M = 1$.

Although not shown here, we have also computed the performance for the case where the magnitudes of the disorder are small, namely $-20, -23, -26$, and -30 dB. The results are similar for this case with the following exceptions. First, the expected delay is much greater

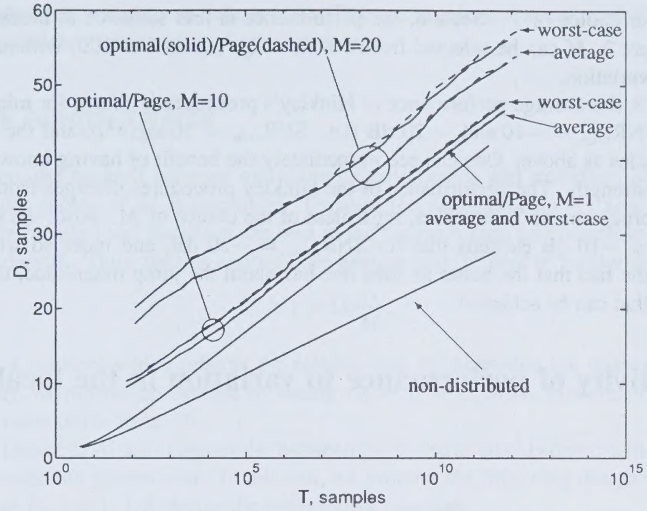


Figure 2: Average and worst case performance of the ML optimal, Page, and non-distributed procedures for $L = 4$ sensors. The jump magnitudes are 0, −3, −6, and − 10dB.

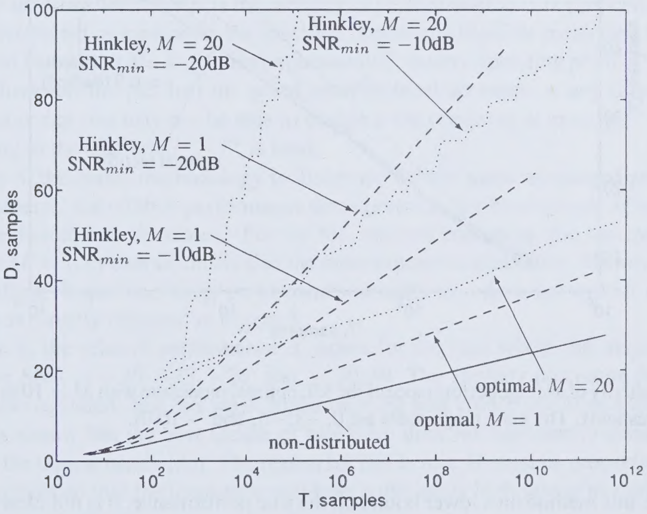


Figure 3: Average performance of Hinkley procedure with $\text{SNR}_{min} = -10$ and − 20 dB, the ML optimal procedure, and the non-distributed procedure for a system with $L = 4$ sensors. The jump magnitudes are 0, −3, −6, and − 10dB.

for a particular choice of T. Second, the performance is less sensitive to blocksize in that, unlike in Figure 2, M can be selected from a wide range (about 1 to 100) without significant performance variation.

In Figure 3, the average performance of Hinkley's procedure is shown for minimum jump magnitudes $\mathrm{SNR}_{min} = -10$ and -20 dB (i.e. $\mathrm{SNR}_{min} = 10\log(\delta^2)$), and the same actual jump magnitudes as above. One can see immediately the benefit of having knowledge of the actual signal strength. The performance of the Hinkley procedures diverges from that of the ML optimal procedure as T increases, regardless of the choice of M. Also, the performance for $\mathrm{SNR}_{min} = -10$ dB exceeds that for $\mathrm{SNR}_{min} = -20$ dB, and more so with larger T. This reflects the fact that the better an idea one has about the jump magnitude, the better the performance that can be achieved.

6 Sensitivity of performance to variation in the local thresholds

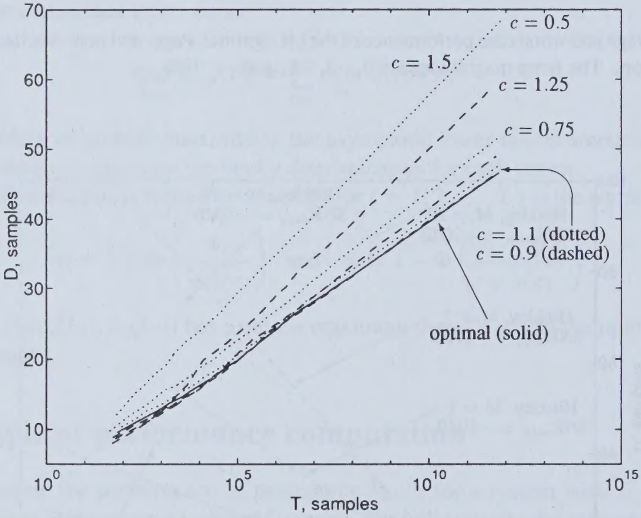


Figure 4: Sensitivity of average performance of the ML optimal procedure with $M = 10$ to perturbations of the local thresholds. The jump magnitudes are 0, −3, −6, and − 10dB.

Since we base this method on a lower bound on the true performance, it is not clear whether the resulting thresholds will lead to reasonable performance. For this reason, a sensitivity analysis is performed as follows. Let $\bar{h}_{loc}$ denote the local thresholds obtained by our method, and let $\bar{h}_{per} = c \cdot \bar{h}_{loc}$ be a perturbation of $\bar{h}_{loc}$, where $c > 0$. Different sets of local thresholds are obtained by choosing different values for c. In Figure 4, the performance is computed for $M = 10$ and $c = 0.5, 0.75, 0.9, 1.0, 1.1, 1.25$, and 1.5. We see that for larger T, the best performance (minimum D) occurred when $c = 1$. Furthermore, the performance suffers noticeably when c is varied by as little as 10%. Thus we conclude that our method for choosing the local thresholds is optimal for practical purposes.

7 Block length effects

In this section, the tradeoff between minimum expected delay and minimum bandwidth cost for a particular choice of M is examined. Because the sampling rate is fixed, the larger M is, the less frequently local decisions are sent to the fusion center, and therefore the lower the required bandwidth. Thus, for this analysis, we assume that the cost of bandwidth is

$$C(M) = C_0 \frac{f_s}{M} \tag{2}$$

where C_0 is a constant which reflects the relative cost of increasing the channel bandwidth. For simplicity, we normalize the cost by letting $C_0 = f_s = 1$. Note, however, that one is not restricted to costs of the form (2).

The best choice of M is a compromise between the desire to quickly detect a disorder and the need to minimize the system cost. To this end, we propose the following design methodology. For procedure $\mathcal{P}_i$, $i = 1, 2, 3$, define the nonnegative function

$$\mathcal{R}_i(T^*, M) \triangleq \min_{T \geq T^*} \frac{D_i(T, M)}{D_0(T^*)} - 1$$

where $D_i(T, M)$ is the expected delay for procedure $\mathcal{P}_i$ when parameters T and M are used, and $D_0(T)$ is the expected delay for the non-distributed procedure. $\mathcal{R}_i(T^*, M)$ is a *relative performance measure* which reflects the *sacrifice* in performance in using procedure $\mathcal{P}_i$ instead of the non-distributed procedure for the specified minimum allowable mean time between false alarms T^* and block size M. $D_0(T)$ is continuous in T, so any operating point T^* is achievable using $\mathcal{P}_0$. However, the fact that the set of possible local decisions at any time is a discrete finite set means that one may not be able to design a test operating at exactly T^*, so the delay corresponding to the smallest $T \geq T^*$ is used.

In Figure 5, the above methodology is illustrated for the cases mentioned previously. We see that in general, the relative performance deteriorates as the blocklength M increases, that is, as the bandwidth cost decreases. For the ML optimal procedure, the best performance is achieved for $M = 1$; of course, this is also the most expensive alternative. However, a designer might be willing to sacrifice some performance in order to reduce the overall cost, and this compromise is directly reflected in Figure 5.

In Figure 6, the relative performance is shown for the case where the magnitudes of the disorder have SNR $= -20, -23, -26$, and -30 dB. The Hinkley procedure for SNR$_{min} = -30$ dB is also included. For this weak signal case the best choice for M is about 100, rather than 1. This shows that the best choice of blocksize does not necessarily correspond to the system with the largest bandwidth. The reason for this is that M must be proportionately larger for weaker signals so that the local tests will have sufficiently high power to enable the global decision to occur quickly.

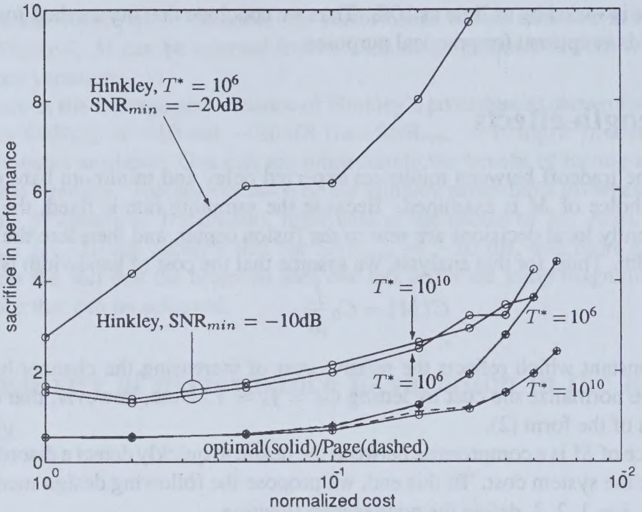


Figure 5: $\mathcal{R}_i(T^*, M)$ versus $C(M)$ for the strong signal case.

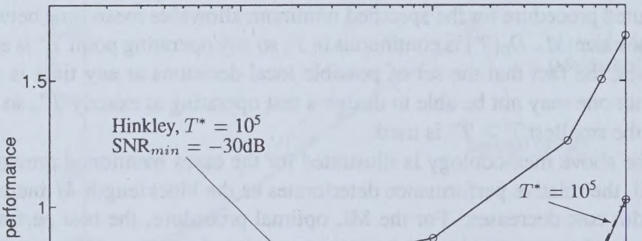


Figure 6: $\mathcal{R}_i(T^*, M)$ versus $C(M)$ for the weak signal case.

8 Conclusions

In this paper, we have examined several alternatives for distributed quickest detection, each of which was shown to have a recursive implementation. It was shown that the performance of the ML optimal and suboptimal procedures is practically the same, and so in practice the latter test might be the better alternative. A simple method for choosing the thresholds of the local detectors based upon a lower bound on the asymptotic performance measure was introduced, and a sensitivity analysis revealed that the resulting thresholds are optimal for practical purposes. Finally, a methodology was devised to determine the best choice of blocklength for the local tests. It was seen that in order to minimize the expected delay in detection, the blocksize of the local tests must be larger when the jump magnitudes are smaller.

References

[1] D. Brook and D. A. Evans, "An approach to the probability distribution of a cusum run length," *Biometrika*, vol. 59, no. 3, pp. 539-549, 1972.

[2] R. W. Crow, *Ph.D. Thesis*, Princeton University, in preparation.

[3] H. R. Hashemi and I. B. Rhodes, "Decentralized sequential detection," *IEEE Trans. Inform. Theory*, vol. IT-35, no. 3, May 1989.

[4] M. Kam, Q. Zhu, and W. S. Gray, "Optimal data fusion of correlated local decisions in multiple sensor detection systems," *IEEE Trans. Aero. Electron. Sys.*, vol. AES-28, no. 3, Jul. 1992.

[5] G. Lorden, "Procedures for reacting to a change in distribution," *Ann. Math. Statist.*, vol. 42, no. 6, pp. 1897-1908, 1971.

[6] E. S. Page, "Continuous inspection schemes," *Biometrika*, vol. 41, pp. 100-114, 1954.

[7] A. R. Reibman and L. W. Nolte, "Optimal detection and performance of distributed sensor systems," *IEEE Trans. Aero. Electron. Sys.*, vol. AES-23, no. 1, Jan. 1987.

[8] D. Teneketzis, "The decentralized quickest detection problem," *Proc. 21st IEEE Conf. on Decision and Contr.*, Orlando, FL, Dec. 1982, pp. 673-679.

[9] D. Teneketzis and P. Varaiya, "The decentralized quickest detection problem," *IEEE Trans. Auto. Control*, vol. AC-29, no. 7, Jul. 1984.

[10] R. R. Tenney and N. R. Sandell, Jr., "Detection with distributed sensors," *IEEE Trans. Aero. Electron. Sys.*, vol. AES-17, no. 4, Jul. 1981.

[11] V. V. Veeravalli, T. Basar, and H. V. Poor, "Decentralized sequential detection with a fusion center performing the sequential test," *IEEE Trans. Inform. Theory*, vol. IT-39, no. 2, Mar. 1993.

A Lower Bound on the Probability of Error in Multihypothesis Testing [*]

H. Vincent Poor and Sergio Verdú[†]
Princeton University

Abstract

Consider two random variables X and Y, where X takes on values in $\{1, 2, \ldots, M\}$, and where Y is arbitrary. Let ϵ denote the minimum probability of error incurred in estimating X from Y. It is shown that

$$\epsilon \geq (1-\alpha) P\left(\pi(X|Y) \leq \alpha\right), \quad 0 \leq \alpha \leq 1,$$

where $\pi(X|Y)$ denotes the posterior probability of X given Y. This bound generalizes a bound due to Shannon, which corresponds to the particular case $\alpha = 1/2$ and equiprobable hypotheses.

Summary

Consider two random variables X and Y, where X takes values in $\{1, 2, \ldots, M\}$, and where Y is arbitrary. The minimum probability of error (or maximum-*a posteriori*) estimate of X given Y is defined by

$$\hat{X} = \arg\left\{\max_{1 \leq k \leq M} \pi(k|Y)\right\}, \tag{1}$$

where

$$\pi(k|Y) \stackrel{\triangle}{=} P\left(X = k|Y\right). \tag{2}$$

Let ϵ denote the probability of error incurred by (1); i.e.,

$$\epsilon = P\left(\hat{X} \neq X\right).$$

We then have the following result (which holds even if M is not finite).

Proposition: For each $\alpha \in [0, 1]$, we have

$$\epsilon \geq (1-\alpha) P\left(\pi(X|Y) \leq \alpha\right). \tag{3}$$

---

[*] This work was supported by the U. S. Army Research Office under Grant DAAH04-93-G-0219.

[†] Mailing Address: Department of Electrical Engineering, Princeton University, Princeton, NJ 08544.

Proof: Let $Z_1, Z_2, \ldots, Z_M$, denote the random variables $\pi(1|Y), \pi(2|Y), \ldots, \pi(M|Y)$, placed in decreasing order, and note that

$$\epsilon = 1 - E\left\{\pi(\hat{X}|Y)\right\} = 1 - E\left\{Z_1\right\}. \tag{4}$$

For any $\alpha \in [0,1]$, we can write

$$P\left(\pi(X|Y) > \alpha\right) = E\left\{\sum_{k=1}^{M} \pi(k|Y) 1\{\pi(k|Y) > \alpha\}\right\}, \tag{5}$$

where $1\{A\}$ denotes the indicator function of the event A. The argument of the expected value in (5) can be written as

$$\sum_{k=1}^{M} \pi(k|Y) 1\{\pi(k|Y) > \alpha\} = \sum_{k=1}^{M} Z_k 1\{Z_k > \alpha\}, \tag{6}$$

from which it follows that

$$P\left(\pi(X|Y) > \alpha\right) \geq E\left\{Z_1 1\{Z_1 > \alpha\}\right\}. \tag{7}$$

Note that, for any $\alpha \in [0,1]$ we have

$$Z_1 = \alpha Z_1 + (1-\alpha) Z_1 \leq \alpha + (1-\alpha) Z_1 1\{Z_1 > \alpha\}, \tag{8}$$

since $Z_1 \leq 1$. Thus, on combining (7) and (8), we have

$$E\left\{Z_1\right\} \leq \alpha + (1-\alpha) E\left\{Z_1 1\{Z_1 > \alpha\}\right\} \leq \alpha + (1-\alpha) P\left(\pi(X|Y) > \alpha\right), \tag{9}$$

which, together with (4), implies the bound (3). □

The above bound generalizes Shannon's lower bound on ϵ, which is given by the particular case of (3) with $\alpha = 1/2$ and X equiprobable at $\{1, \ldots M\}$. (See [1].) It can be shown by example that the bound (3) is generally not maximized at $\alpha = 1/2$, and thus it provides tighter bounding of the error probability than does Shannon's bound. The bound (3) also tightens the bound of Verdú and Han [2], which is given by

$$\epsilon \geq P\left(\pi(X|Y) \leq \alpha\right) - \alpha, \quad 0 \leq \alpha \leq 1. \tag{10}$$

The bound (10) is used in [2] to provide a converse to the channel coding theorem, which leads to a general formula for channel capacity. Note that the bound (10) is smaller than (3) by the amount

$$\alpha P\left(\pi(X|Y) > \alpha\right). \tag{11}$$

Like Shannon's bound, the Verdú-Han bound was developed under the assumption of equiprobable hypotheses.

[1] C. E. Shannon, "Certain results in coding theory for noisy channels," *Information and Control*, Vol. 1, pp. 6-25, September 1957.

[2] S. Verdú and T. S. Han, "A general formula for channel capacity," *IEEE Trans. Information Theory*, to appear, 1993.

Progress Toward Optimal Nonblocking Multipoint Virtual Circuit Switching Networks

Jonathan S. Turner
Computer Science Department
Washington University, St. Louis

Abstract

Over the last several years, a variety of multipoint virtual circuit switching systems have been proposed for use in Asynchronous Transfer Mode (ATM) networks. The inherent flexibility of ATM networking and the desire to apply this technology to a wide range of different applications makes it impractical to rely on statistical traffic models when engineering a system. This motivates a renewed interest in networks which are nonblocking with respect to requests to create or modify multipoint virtual circuits. While the various multipoint switching system architectures can all be configured to achieve nonblocking operation, for most, the cost of the nonblocking configuration quickly becomes prohibitive. We examine the complexity characteristics of a variety of multipoint switch architectures, taking into account the switching network, the memory required for routing multipoint virtual circuits and the effort required to create or modify a virtual circuit. In particular, we show that a very simple architecture, based on the concept of cell recycling, can be configured to be nonblocking with optimal complexity in all three dimensions. This architecture exemplifies a class of networks which we refer to as *reroutably nonblocking*.

1. Introduction

Multipoint virtual circuit networks support communication paths from a sender to an arbitrary number of receivers. Multipoint virtual circuits induce a tree in the network connecting the sender to the receivers. Switching systems participating in the virtual circuit replicate received cells using *virtual circuit identifiers* in the cell headers to access control information stored in the switching system's internal control tables, then use this information to identify the outputs the cells are to be sent to and relabel the copies before forwarding them on to other switching systems. Figure 1 illustrates the function of a multipoint virtual circuit switch. The switch includes control information, shown

---

This work was supported by the National Science Foundation (grant DCI 8600947), Ascom Timeplex, Bell Communications Research, Bell Northern Research, Goldstar Information and Communications, Italtel SIT, NEC, NTT and SynOptics.

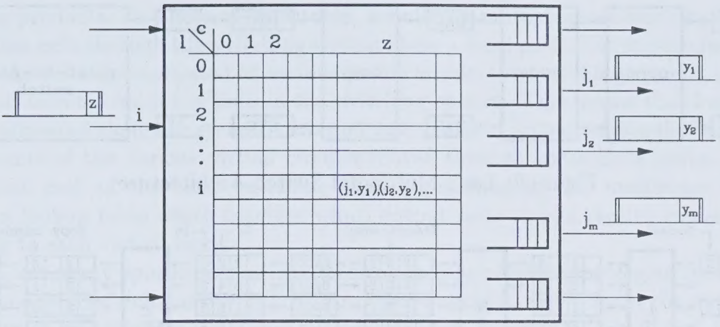


Figure 1: Multipoint Switch Functionality

here as a table, which for each incoming virtual circuit provides a list of outputs and outgoing virtual circuit identifiers. For a cell received on input link i and virtual circuit z, the switch forwards copies to outputs $j_1, j_2, \ldots$ after relabeling them with new virtual circuit identifiers, $y_1, y_2, \ldots$

This paper explores the question of how to design a multipoint virtual circuit switch so that no legitimate request to establish a new virtual circuit or extend an existing virtual circuit (adding an additional endpoint) need be refused for lack of switching system resources. That is, we are interested in systems that are nonblocking with respect to virtual circuit requests and have minimal complexity, where the complexity measures we are concerned with are (1) the hardware required for the switching network, (2) the memory required for defining the collection multipoint virtual circuits and (3) the amount of work that must be done to respond to a requested incremental change to a virtual circuit. For a system with n inputs and outputs and capable of supporting m virtual circuits per input/output, an optimal system will has a network complexity of $O(n \log n)$, contains $O(mn)$ words of memory with $O(\log mn)$ bits per word and can be reconfigured in response to an incremental change request by modifying a fixed number of memory words.

We study this question by reviewing the architecture of several multipoint virtual circuit switching systems, showing how they can be configured to provide nonblocking multipoint operation and determining the complexity of these configurations. We conclude with a novel multipoint switch architecture that is conceptually very simple and is simultaneously optimal in all three complexity dimensions.

2. Lee's Multipoint Switch Architecture

Lee [6] describes a multipoint switch architecture that uses unbuffered switching components and a novel contention resolution scheme to handle the fanout contention associated with multipoint virtual circuits. The design is shown in Figure 2. It includes a point-to-point switching system on the right preceded by several additional components to handle multipoint. Cells are received at the *Port Processors* (PP) on the left where they are assigned a *Fanout* and a *Broadcast Channel Number* (BCN). The cells then pass through a concentrator network, which places the cells on consecutive outputs so as to ensure non-blocking operation of the subsequent networks. Next, the cells pass through a running adder network, which for the cell on output port i computes the sum of the

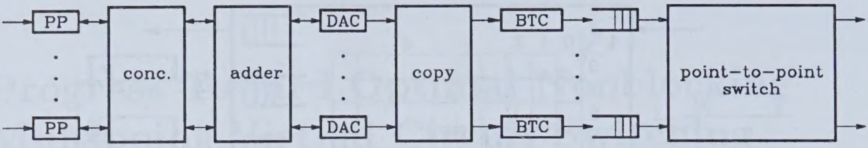

PP
PP
conc.
adder
DAC
DAC
copy
BTC
BTC
point–to–point switch

Figure 2: Lee's Multipoint Switch Architecture

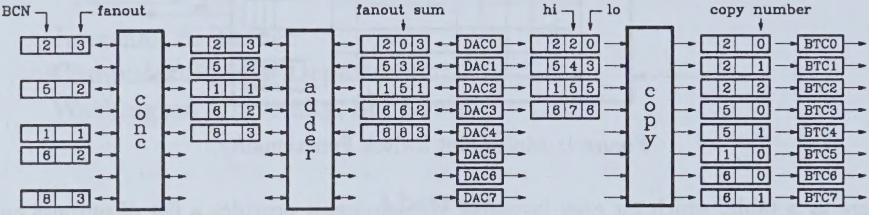

BCN
fanout
2 3
5 2
1 1
6 2
8 3
conc
2 3
5 2
1 1
6 2
8 3
adder
fanout sum
2 0 3
5 3 2
1 5 1
6 6 2
8 8 3
DAC0
DAC1
DAC2
DAC3
DAC4
DAC5
DAC6
DAC7
hi
lo
2 2 0
5 4 3
1 5 5
6 7 6
copy
copy number
2 0
2 1
2 2
5 0
5 1
1 0
6 0
6 1
BTC0
BTC1
BTC2
BTC3
BTC4
BTC5
BTC6
BTC7

Figure 3: Example of Operation of Lee's Multipoint Architecture

fanouts of all cells entering on ports 0 through $i-1$ and places this sum in a field of the cell (this is done for all output ports, using a network with $n \log_2 n$ simple processing elements). Following the adder, the cells enter a set of *dummy address encoders* (DAC), which perform two functions. First, the DAC at output i sets a variable *lo* to be the sum computed by the adder network for output i and lets $hi = lo + f_i - 1$ where f_i is the fanout of the cell at output i. If *hi* exceeds the number of outputs of the entire network, the cell is discarded. The DACs return an acknowledgement to the sending input for each cell that is not discarded. The discarded cells are retransmitted later.

For the cells that are not discarded, *lo* and *hi* are inserted into the cell header as the cell is sent to the copy network. The copy network sends copies of each cell to all the outputs in the range from *lo* to *hi*. The copy network also labels the copies and when each copy reaches a *Broadcast Translation Circuit* (BTC) its broadcast channel number and the copy number are used to produce an output address that the point-to-point switch uses to guide the cell to its ultimate destination. An example, illustrating the operation of Lee's architecture is shown in Figure 3.

While a variety of point-to-point switches can be used in this architecture, the original intent was that it would be used with an unbuffered switching network based on Batcher's bitonic sorter [1]. This leads to a complexity of $O(n(\log n)^2)$ for the switching network. As Lee describes it, the architecture requires $O(mn^3)$ words of memory to support nonblocking operation and requires changes $O(n^2)$ words of memory in order to modify a virtual circuit. However [10] develops a series of improvements which taken together lead to version that can be configured for nonblocking operation with $O(mn^2)$ words of memory. For this configuration, the number of memory words that must be changed to modify a multipoint virtual circuit is $O(n)$.

3. Buffered Multistage Networks with Fixed Path Routing

The Prelude project of the early 1980s [3] introduced a multipoint switch architecture based on a natural extension of classical synchronous time division switching systems. Several manufacturers have adopted variations of this architecture in their current ATM switching products. In these architectures, a buffered multistage network carries user information cells through the switching system along a fixed path. For multipoint virtual circuits, the cells are replicated at various points within the switching system, inducing a tree that uses a fixed set of links in the switching system. This means that bandwidth must be allocated along the system's internal data paths to accommodate the bandwidth requirements of the various virtual circuits routed through those data paths. It also means that each of the basic switching elements making up the multistage network requires a lookup table which specifies which output links should receive copies of cells belonging to each virtual circuit.

There are many specific multistage interconnection network topologies that can be used. The *extended delta networks* [12] form a general class that includes the well-known delta network and the Beneš network as special cases. Let $d \geq 2$, $k \geq 0$, $0 \leq h < k$ be integers and let $n = d^k$. The extended delta network $D^*_{n,d,h}$ is constructed from $d \times d$ switch elements and has $k + h$ stages. The last k stages are topologically equivalent to a d-ary delta network. The initial h stages can be thought of as distribution stages which provide d^h distinct routes between any input and any output. When $h = 0$, the extended delta network is equivalent to the ordinary delta network and when $h = k - 1$, it is equivalent to a Beneš network.

All of these networks have $O(n \log n)$ complexity for fixed d. To achieve nonblocking operation, the speed of the internal data paths must be higher than that of the external links. We denote the ratio of the internal data path speed to the external link rate by σ and refer to it as the *speed advantage* of the system. Since for any fixed clock rate an increase in speed advantage requires increasing data path width, the cost of a system grows with σ giving an overall system cost that is $O(\sigma n \log n)$. In [7], it is shown that for point-to-point virtual circuits it is necessary to have $\sigma \geq 1 + 2(1 + (d-1)(k-1))/d$ when $h = k - 1$ (that is, for the Beneš network, which provides the best performance for this class of networks) which implies that the system complexity is $O(n(\log n)^2)$. For multipoint virtual circuits, the situation is worse. Here, we require a speed advantage of approximately $(n/d)(1 - 1/d)$ to avoid blocking, making the complexity $O(n^2 \log n)$ for any fixed d. Adding an endpoint to an existing virtual circuit requires changing $O(\log n)$ table entries.

To determine the memory requirements we need to consider two options. The conceptually simpler option is to have the lookup table at each switch store a bit vector of d bits (specifying a subset of the switch outputs) for each of the different multipoint virtual circuits that could be simultaneously active ($mn/2$ in the worst-case). This leads to a worst-case memory complexity of $O(mn^2 \log n)$. The second option is to have the lookup table specify not only the output ports to be used for a given multipoint connection but new virtual circuit identifiers for each copy as well. In order for each of the internal data paths to accommodate m virtual circuits, the total number of table entries would be $O(mnk)$, and the size of each entry would be $d \log m$ bits instead of a single bit. This leads to a complexity of $O(mn \log n)$ words. We note that these memory estimates underestimate the real system complexity since the required speed advantage would force a more costly memory organization as well. This can increase the real cost of the lookup tables by another factor of n.

In [8], it is shown that by cascading two Beneš networks together, one can achieve nonblocking operation for new virtual circuits (but not necessarily for extending an existing virtual circuit) with the same speed advantage required for point-to-point virtual circuits. This means that by rearranging just the virtual circuit we are seeking to extend, it is always possible to add a new endpoint. We use the term *reroutably nonblocking* for networks for which it is possible to avoid blocking in this way. The amount of memory required for this network is roughly double that needed for the single Beneš network. The number of words that must be changed in the worst-case when adding an endpoint to a virtual circuit with fanout f is $O(f \log n)$, including the time needed for rerouting. If no rerouting is required, the time is $O(\log n)$.

4. Buffered Multistage Networks with Per Cell Routing

Buffered multistage switching networks can also be constructed so that cells are routed individually, allowing different cells in a given virtual circuit to take different paths through the switching network. This means that cells may exit the network in a different order from that by which they entered, requiring a resequencing buffer at each output port of the network. The depth of the resequencing buffers must be proportional to the network delay to ensure that cells that pass through the network quickly can be held up long enough to allow cells that take a longer time to get through a chance to catch up. Since delay grows in proportion to the number of stages in a buffered multistage network, this does not change the growth rate of the system complexity, so we need not consider the resequencing cost further.

Networks which route cells independently, distribute the traffic as evenly as possible across the switching network's internal data paths allowing the system to operate with a much smaller speed advantage that when fixed path routing is used. To determine the necessary speed advantage, we can apply an analytical technique developed in [9]. We denote a virtual circuit by a triple (x, Y, ω), where x is an input to the network, Y is a set of outputs and $0 < \omega < 1$ is a *weight*, which represents the data rate of the virtual circuit, taken as a fraction of the network's internal data path speed.

We say that a virtual circuit *induces a load* on the links that lie on paths joining the virtual circuit's input and output ports. In particular, we assume that the load is distributed evenly whenever there are multiple paths to the desired destination. We let $0 < \alpha \leq 1$ denote the maximum load allowed on any of the network's input or output ports. A *virtual circuit assignment* C is a set of virtual circuits that satisfies this constraint and the load induced by C on link ℓ is denoted $\lambda_\ell(C)$. We say a network with per cell routing is nonblocking if for all virtual circuit assignments, $\lambda_\ell(C) \leq 1$ for all links ℓ.

A network with the extended delta topology can be used for switching multipoint virtual circuits by having it perform traffic distribution in the first h stages and then copying the cells to the required outputs in the final k stages. Hence, for $h \geq 1$, if a virtual circuit places a load ω on some input x of the network, the load on the links exiting from x's first stage switch is $1/d$, the load on the links exiting from the subsequent second stage switches is $1/d^2$ and so forth. This spreading of the load continues for the first h stages and then stops. In the last h stages, the load builds up again, increasing by a factor of d at every stage. Let ℓ be a link in stage i of an extended delta network and let $c = (x, Y, \omega)$ be a virtual circuit. From the above discussion, if there is a path in the network from x to ℓ and a path from ℓ to any output in Y, then

$$\lambda_\ell(c) = \begin{cases} \omega_j d^{-i} & 0 \leq i \leq h \\ \omega_j d^{-h} & h \leq i \leq k \\ \omega_j d^{-(k+h-i)} & k \leq i \leq k+h \end{cases}$$

Let C be any set of virtual circuits on $D^*_{n,d,h}$ and let C_ℓ be the subset of virtual circuits $c_j = (x_j, Y_j, \omega_j)$ for which there is a path from x to ℓ and a path from ℓ to some output in Y. Note that there are paths to ℓ from at most d^i inputs and from ℓ to at most d^{h+k-i} outputs. Because the load on every input and output is limited to α,

$$\sum_{c_j \in C_\ell} \omega_j \leq \alpha d^i \quad \text{and} \quad \sum_{c_j \in C_\ell} \omega_j \leq \alpha d^{k+h-i}$$

Thus, for $0 \leq i \leq h$,

$$\lambda_\ell(C) = \sum_{c_j \in C_\ell} \lambda_\ell(c_j) \leq d^{-i} \sum_{c_j \in C_\ell} \omega_j \leq \alpha$$

For $k \leq i \leq k+h$,

$$\lambda_\ell(C) = \sum_{c_j \in C_\ell} \lambda_\ell(c_j) \leq d^{-(k+h-i)} \sum_{c_j \in C_\ell} \omega_j \leq \alpha$$

For $h \leq i \leq k$,

$$\begin{aligned} \lambda_\ell(C) &\leq d^{-h} \sum_{c_j \in C_\ell} \omega_j \leq \alpha d^{-h} \max_{h \leq r \leq k} \min\left\{d^r, d^{k+h-r}\right\} \\ &= \alpha d^{-h} d^{\lfloor (k+h)/2 \rfloor} = \alpha d^{\lfloor (k-h)/2 \rfloor} \end{aligned}$$

Hence, for any assignment C on $D^*_{n,d,h}$, $\lambda_\ell(C) \leq \alpha d^{\lfloor (k-h)/2 \rfloor}$ for all links ℓ. For $h = k-1$, this is simply $\lambda_\ell(C) \leq \alpha$. That is, the load on the internal links of the network is bounded by the load on the inputs and outputs. Hence, the only speedup needed in the network is that required to accommodate queueing effects. From [2] it is apparent that if the network is constructed from shared buffer switches with $d \geq 8$ and $4d$ buffer slots, a speed advantage as low as 20% can suffice.

This analysis implies that the switching network complexity needed to achieve nonblocking operation is $O(n \log n)$. As in fixed path networks, lookup tables are needed in at least the last k stages of the network. Because cells are routed independently, the size of the routing tables must be larger. In particular, the number of memory words required for nonblocking operation increases to $O(mn^2 \log n)$. The number of words that must be changed to modify a virtual circuit is large but the number of control cells needed to effect this change can be reduced by having the network copy the control cells. Taking this into account, the effective cost of modifying a virtual circuit is $O(\log n)$.

5. Broadcast Packet Switch

The broadcast packet switch architecture [10] is a multipoint switch architecture which is also based on the principle of per cell routing. There is a variety of specific design choices one can make with this architecture, but stated in a general way, it consists of a pair of extended delta networks, cascaded one after the other. The first network (called the copy network) is responsible for making copies of cells belonging to multipoint virtual circuits, using a fanout field in the cell header to produce the appropriate number of copies. The second network (called the routing network) is responsible for routing the individual copies to the proper network outputs. In between the two sections is a set of lookup tables which assign each of the copies an outgoing link number. The networks themselves do not require any lookup tables but rely entirely on information carried in the cell headers. By an analysis similar to the one given above, it has been shown that the networks require only a small constant speed advantage (with the exact speed advantage depending on the number of traffic distribution stages in the copy and routing networks) [7]. Hence, the network complexity is $O(n \log n)$.

Each of the n lookup tables requires an entry for each of the mn multipoint virtual circuits and each entry stores information relating to a single output of a virtual circuit. This means that the memory requirement is $O(mn^2)$ and $O(n)$ words may need to be changed to modify a virtual circuit. By having the copy network replicate control cells, the number of control cells that must be sent to effect this change can be reduced to $O(f)$ for virtual circuits with fanout f.

The size of the lookup tables can be dramatically reduced if the copying process is modified so that cells with smaller fanouts are delivered only to a subset of the lookup tables. Let $f_0 = 1 < f_1 < f_2 < \cdots < f_r = n$. We assign a virtual circuit to *fanout class i* if its fanout exceeds f_{i-1} but not f_i. If the copy network routes virtual circuits in class i to one of a set of nonoverlapping ranges of f_i outputs, the number of multipoint virtual circuits of class i that can be handled with a given amount of memory can be increased by a factor of n/f_i. If i maximizes $f_i/(f_{i-1}+1)$, all virtual circuits have fanout $f_{i-1}+1$ in the worst-case. In this case, we can have $mn/(f_{i-1}+1)$ different virtual circuits and we use a total of f_i lookup table entries for each one. To handle the worst-case then, $mn(f_i/f_{i-1})$ total entries are enough. (In the original architecture, there is only one class and we need $mn^2/2$ table entries to handle the worst-case situation.) If f_i/f_{i-1} is the same for all i, then the number of classes is $\log_{(f_i/f_{i-1})} n$.

Thus, we can cut the amount of memory needed to $2mn$, using $\lg n$ levels (where lg denotes the base 2 logarithm). Unfortunately, fanout f virtual circuits now require consecutive sets of lookup tables with sufficient unused bandwidth. To prevent excessive loading at the lookup tables, an additional speed advantage is needed which grows in proportion to the number of fanout classes. The trade-off between the number of table entries and the network complexity (resulting from the increased speed advantage) is illustrated for several examples in the following table.

r	table entries	network complexity
1	$O(mn^2)$	$O(n \log_d n)$
2	$O(mn^{3/2})$	$O(n \log_d n)$
$\lg\lg n$	$O(mn^{1+1/\lg\lg n})$	$O(n(\log_d n)(\lg\lg n))$
$\lg n$	$O(mn)$	$O(n(\log_d n)(\lg n))$

For $n = 2^{16}$, $\lg\lg n$ is 4 and $n^{1/\lg\lg n}$ is 16.

6. Multipoint Switching with Cell Recycling

Another approach for switching multipoint virtual circuits is based on the concept of *cell recycling* [10]. This is illustrated in Figure 4. To implement a multipoint virtual circuit, a binary tree is constructed with the source port at its root and the destination ports at its leaves. Internal nodes represent ports acting as relay points, which accept cells from the switching network but then recycle them back into the network after relabeling the cells with a destination pair identifying the next two ports they are to be sent to. In this example, a multipoint virtual circuit delivers cells from input a to outputs b, c, d and e, using ports x and y as relay points. In the lower part of the diagram, the implementation of the virtual circuit is shown in an 'unrolled' form, to clarify the flow of cells through the system. It should be understood however, that this is purely illustrative. There is in

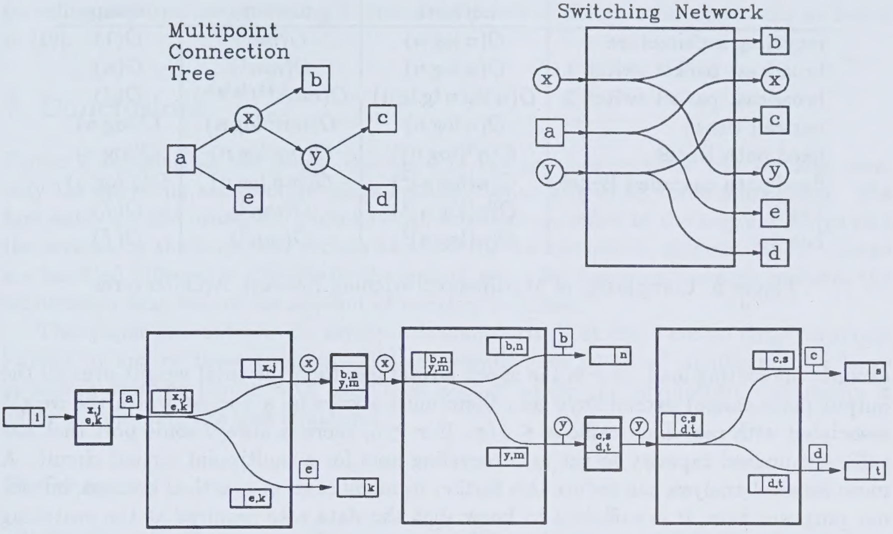


Figure 4: Multipoint Switching by Recycling Cells

fact just one switching network, not three, and cells are simply sent through it multiple times in order to reach all the destinations. In the example, cells entering at input a with virtual circuit identifier (VCI) i, are forwarded to output e, VCI k and output x, VCI j. At x, the cell is recycled, with VCI j used to select a new table entry from x's VXT. The resulting information causes the cell to be forwarded to output b, VCI n and output y, VCI m. At y, the cell is recycled again, with the resulting copies delivered to c and d. Note that all the information needed by the switching network is placed in the cell header by the lookup table at the switch port, eliminating the need for any lookup tables within the switching network.

To add an endpoint to a multipoint virtual circuit, some rerouting of the virtual circuit is needed. In the example shown in Figure 4, adding a new output f, could be done by replacing e with a new internal node z and making e and f children of z. In general, some leaf in the tree is replaced with a new internal node and the old leaf and a new leaf (correponding to the new output) are made children of the new internal node. The only constraint on the selection of the leaf to be replaced is that we avoid increasing the depth of the tree unnecessarily. The only constraint on the choice of a new internal node is that it correspond to a port with sufficient unused capacity on the recycling data path to accommodate the recycled virtual circuit. Note that adding an endpoint requires changing only two table entries. Dropping an endpoint is similar. The only special requirement is that we restructure the tree as necessary so that every internal node has two children. This can be accomplished

by changing a single table entry.

To determine the switching network cost, we need to consider the impact of the recycling strategy on the total traffic in the network. A binary tree with r leaves and in which every internal node has two children, has exactly $r - 1$ internal nodes. Hence, a multipoint virtual circuit with rate ω and r leaves places a total load of ωr on the outgoing links and generates a recycling load of $(r-2)\omega$. Thus, the recycling load never

	network	memory	setup
recycling architecture	$O(n \log n)$	$O(mn)$	$O(1)$
broadcast packet switch 1	$O(n \log n)$	$O(mn^2)$	$O(n)$
broadcast packet switch 2	$O(n \log n \lg\lg n)$	$O(mn^{1+1/\lg\lg n})$	$O(f)$
per cell Beneš	$O(n \log n)$	$O(mn^2 \log n)$	$O(\log n)$
fixed path Beneš	$O(n^2(\log n))$	$O(mn \log n)$	$O(\log n)$
fixed path cascaded Beneš	$O(n(\log n)^2)$	$O(mn \log n)$	$O(f \log n)$
Lee 1	$O(n(\log n)^2)$	$O(mn^3)$	$O(f)$
Lee 2	$O(n(\log n)^2)$	$O(mn^2)$	$O(f)$

Figure 5: Complexity of Multipoint Switching Network Architectures

exceeds the exiting load. If σ is the speed advantage, then the total weight over all the output ports cannot exceed $2n/\sigma$ and there must always be a port at which the weight associated with recycling traffic is $\leq 1/\sigma$. If $\sigma \geq 3$, there is always some port that has sufficient unused capacity to act as a recycling port for a multipoint virtual circuit. A more detailed analysis can reduce this further in many cases of practical interest, but for our purposes here, it is sufficient to know that the data rate required at the switching network ports is only a constant times the data rate of the external links. This means that if we use a Beneš network with per cell routing, the network complexity is $O(n \log n)$.

The amount of memory needed in the lookup tables is determined by the same analysis used to bound the recycling traffic. Because the number of internal nodes in a binary tree is smaller than the number of outputs, the number of table entries used for a multipoint virtual circuit is bounded by its fanout, implying that the amount of memory needed to make the system nonblocking is $O(mn)$.

To keep cells in order, a resequencer is needed at the output of the Beneš network. Cells pass through the resequencer on every pass through the network. This raises one subtle issue when an endpoint is dropped from a virtual circuit. This reduces the number of passes that some cells take through the network by one, which means that cells passing over this virtual circuit immediately before and after the transition can become misordered if the transition is not handled properly. To ensure that cell ordering is maintained, cells taking the shorter path must be delayed for up to twice the usual amount of time, to give cells on the longer path time to catch up. A mechanism implementing this is described in [10]. From the system complexity standpoint, the main impact is to double the size of the resequencer over what would be required otherwise.

Hence the recycling approach results in a network that achieves optimal complexity in terms of the network, the amount of memory required and the number of words that must be changed to modify a multipoint virtual circuit. However, there is of course an added delay involved in the recycling architecture since cells must pass through the network up to $\lg F$ times, where F is the maximum fanout. For $F = n$, this results in an $O((\log n)^2)$ delay which is clearly not optimal. While in some contexts, this delay could be an issue, it is shown in [10] that for practical systems with $n = F = 2^{16}$, the worst-case delay associated with recycling can be kept below 500 μs, making it smaller than the typcial delay in a digital time division circuit switch and completely negligible relative to the transmission delays associated with wide area network applications. It should also be noted that the recycling approach leads to neither a strictly or wide-sense nonblocking system, but rather a reroutably nonblocking system. A more detailed account of the recycling architecture, including a number of practical design refinements can be found in [10].

7. Conclusions

Figure 5 summarizes the results discussed in the previous sections. As we have seen, only the recycling architecture offers optimal complexity in all three dimensions. The first entry for the broadcast packet switch architecture refers to the original design and the second, to the improved version in which virtual circuits in different fanout classes are handled differently. Similarly, the second entry for Lee's architecture includes the refinements that reduce the amount of memory required.

This paper concentrates on asymptotic complexity, but there are no large constants lurking in any of these analyses. Tthese results have practical significance for large switching systems (those with over 1000 ports), where the asymptotic complexity is likely to dominate the cost equation.

References

[1] Batcher, K. E. "Sorting Networks and Their Applications," *Proceedings of the Spring Joint Computer Conference*, 1968, 307–314.

[2] Bianchi, Giuseppe and Jonathan Turner. "Improved Queueing Analysis of Shared Buffer Switching Networks," *Proceedings of Infocom*, March 1993.

[3] Coudreuse, J. P. and M. Servel. "Prelude: An Asynchronous Time-Division Switched Network," *International Communications Conference*, 1987.

[4] Lee, Tony T. "Non-Blocking Copy Networks for Multipoint Packet Switching," *IEEE Journal on Selected Areas in Communications*, 1455–1467, 12/88.

[5] Melen, Riccardo and Jonathan S. Turner. "Nonblocking Multirate Networks," *SIAM Journal on Computing*, 4/89.

[6] Melen, Riccardo and Jonathan Turner. "Nonblocking Multirate Distribution Networks," *IEEE Transactions on Communications*, 2/93.

[7] Turner, Jonathan S, "Fluid Flow Loading Analysis of Packet Switching Networks," *Proceedings of the International Teletraffic Congress*, June 1988.

[8] Turner, Jonathan S., "Design of a Broadcast Packet Network," *IEEE Transactions on Communications*, June, 1988.

[9] Turner, Jonathan S., "Resequencing Cells in an ATM Switch," Technical Report WUCS-91-21, Department of Computer Science, Washington University, St. Louis, Missouri, 1991.

[10] Turner, Jonathan S., "An Optimal Nonblocking Multipoint Virtual Circuit Switch," Technical Report WUCS-93-30, Department of Computer Science, Washington University, St. Louis, Missouri, 1993.

The Design and Analysis of Electronic and Optical High-Performance Packet (ATM) Switches

Mark J. Karol
Broadband Systems Research Department
AT&T Bell Laboratories
101 Crawfords Corner Road, 4F-529
Holmdel, NJ 07733-3030

Abstract

In this paper, we address some key issues in the design of high-performance packet (ATM) switches, which have low latency, low packet-loss probability, and high throughput. We consider both electronic and optical implementations. Primarily, we focus on the impact queueing plays in the performance and design of high-performance packet switches.

1. Introduction

Asynchronous Transfer Mode (ATM) is a promising technology for large-scale deployment in next-generation broadband networks. It offers a potential for tremendous service flexibility not easily obtainable in traditional circuit-switched networks. However, it also presents significant research challenges in switching and networking. In this paper, we address some key issues in the design of high-performance packet (ATM) switches, which have low latency, low packet-loss probability, and high throughput. We consider both electronic and optical implementations. Primarily, we focus on the impact queueing plays in the performance and design of high-performance packet switches.

After some brief background material on queueing in high-performance packet switches, we will first focus on a research effort currently underway at AT&T to demonstrate an ATM switch architecture scalable to a terabit-per-second capacity (e.g., a 400x400 switch with a transmission rate of 2.5 Gb/s per port).* This implementation is based on a Growable Switch Architecture, which consists of a memoryless front-end Cell Distribution Network followed by a column of small-size Output ATM Switch Modules. It yields optimal delay-throughput performance for arbitrary traffic patterns when scaled to arbitrarily large sizes. For practical deployment, it is advantageous to implement the Output ATM Switch Module as a cascaded concentrator plus a small-size, shared-memory ATM switch. This Concentrator-Based Growable ATM Switch Architecture has the attractive feature that for a negligible impact on performance, the Cell Distribution Network is reduced to a simple broadcast network, and the Output ATM Switch Module can be deployed modularly as an initial small ATM switch and a subsequent addition of a concentrator.

Next, we address some new features and characteristics of queueing in optical packet switches. All of the fundamental results (e.g., the performance advantages of output queueing and the memory reduction of buffer sharing) still apply to optical packet switches. However, the limited state of optical technology (e.g., the lack of an optical random access memory)

---

* This project is supported in part by ARPA.

puts constraints on the architecture of optical packet switches. The lack of an optical random access memory led many in the past to propose hybrid switch architectures that exploit the advantages of both optics and electronics, using electronics for the queueing of packets. Here, we instead focus our attention on the fundamental performance limitations associated with "all-optical" packet switches, in which the packet buffering is implemented via fiber delay lines, which only temporarily hold packets (in contrast to conventional electronic buffers that can "permanently" buffer packets).

Finally, we propose and analyze an optical packet (ATM) switch architecture that comes close to achieving the optimal performance (i.e., best possible delay-throughput performance and minimal possible buffer requirements) of a random-access, shared-memory design. The proposed Shared-Memory Optical Packet (SMOP) Switch buffers packets in recirculation delay lines of appropriately-selected lengths, and uses a novel control algorithm that: (i) keeps packets in their proper first-in, first-out sequence, (ii) supports multiple levels of priority traffic, (iii) minimizes the needed number of recirculation loops (which reduces the size of the switch fabric), and (iv) ensures that packets pass through the recirculation delay lines only a small number of times (e.g., we present and analyze one example of a control algorithm that guarantees packets pass through the recirculation lines *at most two times*).

2. *Queueing in High-Performance Packet Switches*

Because of the unscheduled nature of arrivals to a packet switch, two or more packets may simultaneously arrive on different inputs destined for the same output. (In this paper, we will interchangeably use both the generic term "packets" and the ATM-specific term "cells.") The switch architecture may allow one of these packets to pass through to the output, but the others must be queued for later transmissions. In recent years, there has been an increased awareness of the impact queueing disciplines play in high-performance packet switches. For instance, the performance advantages of output queueing [1,2], memory sharing [2-4], and trunking (i.e., input or output grouping, or both) are well known. Output queueing achieves the best possible delay-throughput performance because packets are delayed only by the unavoidable congestion caused by two or more packets simultaneously arriving on different inputs destined for the same output. Completely-shared buffering also achieves this optimal delay-throughput performance. In addition, it saves on the total amount of buffering needed to achieve a desired packet loss probability, because, due to the statistical nature of packet arrivals, more efficient use is made of nb buffer locations when they are shared by all outputs, rather than partitioned into b locations for each of the n outputs. These techniques especially become important as one considers "real" traffic that is "bursty" [5-7], which places more stringent demands on packet switches than the usual random (Bernoulli) traffic models.

3. *Growable Switch Architecture*

A Growable Switch Architecture [8-11] has been shown previously to yield optimal delay-throughput performance for arbitrary traffic patterns when scaled to arbitrarily large sizes. It consists of a front-end *memoryless* Cell Distribution Network and a column of Output Packet (ATM) Switch Modules (see Fig. 1). The N outputs are divided into groups of n lines each. All incoming cells are routed through the Cell Distribution Network for instantaneous delivery based on their destination output *group* addresses. As such, the Cell Distribution Network can be memoryless and must perform its routing function for all the cells arriving in each time slot. In other words, in each time slot, all the cells are completely cleared out of the Cell Distribution Network to the respective Output ATM Switch Modules. Since the outputs are grouped into n lines each, there are a total of N/n output groups.

For each output group, the corresponding Output ATM Switch Module has m ($m \geq n$) inputs, meaning that up to m cells can be accepted for that output group in each time slot. Since up to N cells can arrive simultaneously for a particular output group, the maximum (sensible) design range of m is thus $n \leq m \leq N$. A few characteristics of this arrangement need to be emphasized: (i) elimination of memory in the Cell Distribution Network avoids multiple stages of queueing delay to the cells, thus achieving perfect output queueing in the Output ATM Switch Modules; (ii) time-correlated (bursty) cell arrivals *only* affect the sizing of the output buffer in each Output ATM Switch Module, and not the design of the Cell Distribution Network; (iii) the dimension ($m \times n$) of the Output ATM Switch Module can be designed for $N = \infty$ and will thus remain constant for infinite switch growth; and (iv) most important of all, this is a growable architecture that satisfies the I/O constraints of circuit modules.

Notice that the dimension of the Output ATM Switch Module is asymmetric, namely $m \times n$ ($m > n$). However, upon initial deployment of a switch, only one such module may be needed. In such a case, the module has to be designed as a stand-alone switch. As a stand-alone switch, there should be an equal number of input and output ports. Therefore, an $n \times n$ switch module makes sense in a stand-alone initial configuration, and an upgrade to the larger $m \times n$ ($m > n$) arrangement is required for further switch expansion. We will call the $n \times n$ module simply an $n \times n$ ATM Switch, and its upgrade to $m \times n$ as the $m \times n$ Output Packet (ATM) Switch Module. A convenient way to carry out this upgrade is to cascade an $m \times n$ Concentrator in front of an $n \times n$ ATM Switch to form an $m \times n$ Output ATM Switch Module as needed (Fig. 2). The Concentrator accepts only valid cells destined for its dedicated group of output ports on a FIFO basis. In other words, the Concentrator is essentially an m-input n-output FIFO buffer, funneling all its incoming cells into the n outputs on a FIFO basis. This can be designed using the shared-memory approach with a simplified buffer control. The remaining task of routing the cells to specific output ports is done in the $n \times n$ ATM Switch Module.

The most important aspect of this concentrator-based arrangement is that it essentially performs the same as a single-stage Output ATM Switch Module [10,11]. A key question is how large the concentrator-based architecture can grow before performance and implementation become a limiting factor. Our analysis shows that for a large configuration of up to 512x512, the concentrator-based approach remains feasible with acceptable performance. Currently, we are pursuing a research effort to demonstrate an ATM switch architecture scalable to a terabit-per-second capacity (i.e., a 400x400 switch with a transmission rate of 2.5 Gb/s per port).

Hierarchical multiplexing is another powerful technique integral to our prototype efforts [12], and also complementary to the Growable Switch Architecture. Illustrated in Fig. 3, hierarchical multiplexing permits a variety of interfaces of different rates to be deployed flexibly because the incoming signals are multiplexed up to a higher speed for switching in an internal high-speed core fabric. The core fabric speed dictates the maximum interface data rate supported by the switch. As signals are multiplexed inside the switch to higher speeds, say $k{:}1$, the required dimension of the core fabric is reduced to $N/k \times N/k$, but the operating speed is increased to k-times the external line speed. In the context of the Growable Switch Architecture, note that the Cell Distribution Network design becomes easier as the maximum fabric dimension is reduced (i.e., from $N \times N$ to $N/k \times N/k$).

4. Queueing in Optical Packet Switches

All the fundamental results of queueing in high-performance packet switches (e.g., the performance advantages of output queueing and the memory reduction of buffer sharing) reported previously still apply to optical packet switches. However, the limited state of optical technology (e.g., the lack of an optical random access memory) puts constraints on the architecture of optical packet switches. In optical implementations of packet switches, the buffering will typically be implemented via storage in fiber delay lines. In [13], we investigated some new features and characteristics of queueing in optical packet switches. Before reviewing these performance issues, for reference we will first survey some prior work related to the buffering of optical signals in fiber delay lines.

In 1986, Thompson proposed a photonic time-slot interchanger that uses N recirculation loops each of length equivalent to the duration of one time slot [14]. Directional couplers control whether data is recirculated to the delay-line memory units or sent to the outputs. Also that year, Prucnal, Santoro, and Sehgal proposed an optical time-division multiplexer in which a set of M delay lines of different lengths are employed to appropriately delay optical signals such that they are output during the correct time slots [15]. The M delay lines are connected in parallel between two active electrooptic switches, and a control unit selects the desired fiber optic delay path by appropriately activating the switches.

For resolving contentions in packet-switched systems, Chlamtac proposed using a set of fiber delay lines to implement an Optical Delay-Line Packet Switch [16]. The Optical Delay-Line Packet Switch stored and switched packets in the optical domain to resolve contentions, by redistributing contending packets over time. Thus, the Optical Delay-Line Packet Switch introduced the fundamental QUADRO principle: Queueing of Arrivals (i.e., via delay lines) for Delayed Reception.

Haas' Staggering Switch [17-19] is our final example in this brief survey of some optical-switch architectures that use fiber delay lines for temporarily buffering packets. As in [15], the Staggering Switch uses a set of parallel delay lines of different lengths to appropriately delay the optical signals. However, because it is designed for packet switching, "the Staggering Switch architecture is based on two stages: scheduling and switching. Each one of the stages is a reconfigurably nonblocking switching fabric, implemented with electronically controlled optical devices ($LiNbO_3$, for example). The scheduling stage ($n \times m$) is connected to the switching stage ($m \times n$, where $m \geq n$) by m delay lines, d_i, $i = 1$ to m. The delay of the d_i delay line equals i packets." [17] "Among the salient features of the Staggering Switch are: its transparency, lack of recirculation, and flexibility in operation and in performance." [18] The motivation for avoiding recirculation loops (as used in, for example, [14]) is "the large attenuation of the recirculation lines and, thus, the need for optical amplification." [19]

Assuming optical buffering is implemented via "feed-forward" fiber delay lines (i.e., without recirculation delay lines), we have obtained lower bounds on the amount of buffering needed for various traffic models [13]. Numerical examples of these bounds are included in the next section, where we compare the performance of feed-forward architectures with the performance of a proposed shared-memory optical packet switch. In [13], we also intuitively explained why queueing packets via feed-forward delay lines falls short of the optimal performance of completely-shared buffering, and why it doesn't permit high-priority packets to preempt low-priority packets stored in the queue (i.e., in a delay line).

5. *The Shared-Memory Optical Packet (SMOP) Switch*

In this section we describe an optical packet (ATM) switch architecture that comes close to achieving the optimal performance (i.e., best possible delay-throughput performance and minimal buffer requirements - which are dictated by the packet arrival statistics) of a random-access, shared-memory design. The proposed Shared-Memory Optical Packet (SMOP) Switch [20] buffers packets in recirculation delay lines of appropriately-selected lengths, and uses a novel control algorithm that: (i) keeps packets in their proper first-in, first-out sequence, (ii) supports multiple levels of priority traffic, (iii) minimizes the needed number of recirculation loops (which reduces the size of the switch fabric), and (iv) ensures that packets pass through the recirculation delay lines only a small number of times. In so doing, it alleviates problems with feedback-based architectures cited in the past [18-19] as motivation for feed-forward architectures.

Figure 4 illustrates a block diagram of the proposed $n \times n$ Shared-Memory Optical Packet (SMOP) Switch. It consists of n input fibers, n output fibers, m recirculation delay lines (of appropriately-selected lengths) for buffering packets, an $(n+m)\times(n+m)$ memoryless, rearrangeably non-blocking optical switch (e.g., an optical crossbar switch), and some control circuitry (which reconfigures the memoryless, optical switch on a packet-by-packet basis). The Control is most likely implemented electronically, although a simple control algorithm perhaps can be implemented optically. To determine the appropriate configuration of the memoryless, optical switch, the Control: (i) keeps track of all packets buffered in the recirculation loops , so that it knows which packets will be returning to the switch at what times, and (ii) reads the headers of new packet arrivals by tapping a small fraction of their energy (see Fig. 4).

Figure 4 shows that the m recirculation delay lines have lengths d_1, d_2, ..., and d_m; the fiber lengths are expressed in units equal to the number of packets that they can store end-to-end. Note that a total of $B \triangleq (d_1 + d_2 + \cdots + d_m)$ packets can be stored in the recirculation fibers. We assume a time-slotted system in which, each time slot, up to $(n+m)$ packets may arrive at the switch (i.e., n new arrivals plus m "recirculation packets"). Of these packets, up to n can be transmitted to their appropriate output ports; the rest are "buffered" in the recirculation loops. A packet that is buffered in a loop of length d_i will exit the delay line and return to the switch after d_i time slots. The Control selects which packets to output each time slot, and also assigns the remaining packets to the recirculation loops. In [20] we described two control algorithms as illustrations of the excellent performance that can be achieved with the SMOP Switch architecture. In the next section, we provide some more numerical results to show that superior performance is attainable even with only a small number of recirculations (e.g., only *two*).

The use of recirculation delay lines in this packet switch is reminiscent of (an optical implementation of) the Starlite Switch [21]. However, the SMOP Switch uses a central scheduling algorithm and recirculation delay lines of length greater than one (the Starlite Switch uses many lines of length one), which dramatically reduces the needed number of recirculation loops [2] (and hence, the size of the optical switch) and also reduces the number of recirculations per packet. In addition, by tightly controlling the number of recirculations per packet, the Control keeps the power losses low enough to perhaps eliminate the need for optical amplifiers in the delay lines, or, more likely, minimizes the added noise with amplifiers in the recirculation loops. Also, if the power budget does indicate the need for amplifiers in the feedback loops, then the reduction in the number of loops translates into a valuable reduction in the number of amplifiers. In fact, our performance results so far indicate that the number of feedback loops will be approximately equal to n, the number of input (output) ports. And if all packets are sent through the feedback loops (with amplifiers) at least once, there may not be the need for amplifiers at the n input/output ports of the switch.

Finally, before examining the performance of the SMOP Switch, note that buffering packets in recirculation loops offers flexibility and efficiency not possible with feed-forward delay lines. Suppose, for example, first-in first-out requirements dictate that an arriving packet won't be transmitted for at least k time slots (because other packets are queued for the same output). Then, it is advantageous to store the packet in a recirculation delay line of length k (if possible) and "forget about it for a while." Even if the packet cannot be buffered in a delay line of length k, "optimal performance" may still be possible since there may be many combinations of delay-line lengths that sum to k. For example, an optimal (i.e., minimal) delay of 10 time slots can be attained by buffering a packet in a delay line of length 10, or by buffering it successively in delay lines of length 7 and 3, or successive delay lines of length 6, 1, and 3, etc. Furthermore, recirculation permits "scheduling decisions" to be revised each time a packet returns to the memoryless optical switch from the feedback loops (e.g., to allow for quicker transmission of new higher-priority packets).

6. Performance of the SMOP Switch

We assume fixed-length packets synchronously arrive to an $n \times n$ SMOP Switch in a time-slotted fashion, with the duration of each time slot equal to the transmission time of a packet. We model the packet arrivals on the n inputs by independent and identical Bernoulli processes. That is, in any given time slot, the probability that a packet arrives on a particular input is p; each packet has equal probability $1/n$ of being addressed to any given output, and successive packets are independent. More realistic, bursty, traffic models will further illustrate the importance of memory sharing in packet switches.

Figures 5 and 6 show the packet loss probability for switch sizes $n=8$ and $n=16$, respectively. The results are presented as a function of the number of delay lines m (with delay-line lengths $d_1=1$, $d_2=2$, ..., and $d_m=m$) at a 90% traffic loading ($p=0.9$). The curves labeled "SMOP Switch" were obtained using the FIFO Control detailed in [13], in which no packet ever required more than 10 recirculations. The "2 Recirculations (Max)," "3 Recirc." and "4 Recirc." curves were obtained using a Control that limits the maximum number of recirculations per packet to 2, 3, and 4, respectively. As pointed out in [13], including the number of recirculations as a specific part of the Control improves performance. When a packet arrives to the switch it is "scheduled" for transmission in the earliest possible time slot (subject to the FIFO constraint) that can be reached using at most 2 (or 3 or 4) "idle slots" in the set of m delay lines.

For comparison, Figs. 5 and 6 also include a lower bound on the packet loss probability for any feed-forward architecture [13], and a shared-memory lower bound on the loss probability of *any packet switch* that uses m delay lines (with lengths 1, 2, ... , m) for the queueing of packets. This shared-memory bound corresponds to an ideal, completely-shared memory switch, in which any packet arriving to the switch can be stored in any one of the $B=m(m+1)/2$ buffer locations, irrespective of what packets are already stored in which delay lines.

In Figs. 5 and 6, note how close the SMOP performance is to the shared-memory bound. Approximately 13 and 19 delay lines are needed to achieve a 10^{-6} packet loss probability, for $n=8$ and $n=16$, respectively. In contrast, all feed-forward architectures would suffer a packet loss probability greater than 10^{-3} with the same number of delay lines; 49 and 52 delay lines, respectively, would be needed to attain a 10^{-6} loss probability. The difference between the SMOP Switch and feed-forward architectures becomes even more substantial at lower loss probabilities, and with more realistic, bursty traffic models.

As in [13], it still needs to be emphasized that no attempt was made to optimize the Control or the delay-line lengths (we only considered $d_1 = 1,\ d_2 = 2,\ \ldots\ ,\ d_m = m$). Our preliminary studies indicate that providing the required buffering via some longer delay lines further reduces the number of recirculations.

7. Summary

In this paper, we addressed some key issues in the design of high-performance packet (ATM) switches. After some background material on queueing in high-performance packet switches, we first described a large, Concentrator-Based Growable ATM Switch Architecture. Then, turning to "all-optical" implementations of packet switches, we pointed out that queueing packets via feed-forward delay lines falls short of the optimal performance of completely-shared buffering, and that it doesn't permit high-priority packets to preempt low-priority packets stored in the queue. Finally, we presented the fundamental concepts of a new Shared-Memory Optical Packet (ATM) Switch, along with some preliminary performance results to illustrate the key features of this design. In the SMOP Switch, packets are buffered in recirculation delay lines (e.g., fiber loops) of appropriately-selected lengths, and a novel control/scheduling algorithm makes efficient (shared) use of "buffer locations" in the delay lines. Without even attempting to optimize the Control or the delay-line lengths, we showed that the performance of the SMOP Switch can be close to the shared-memory bound.

REFERENCES

[1] M. J. Karol, M. G. Hluchyj, and S. P. Morgan, "Input versus Output Queueing on a Space-Division Packet Switch," *IEEE Trans. Commun.*, vol. COM-35, pp. 1347-1356, Dec. 1987. (Also in *GLOBECOM'86 Conf. Rec.*, pp. 659-665, Dec. 1986.)

[2] M. G. Hluchyj and M. J. Karol, "Queueing in High-Performance Packet Switching," *IEEE Jour. Select. Areas Commun.*, vol. 6, no. 9, pp. 1587-1597, Dec. 1988.

[3] P. Gonet, J. P. Coudreuse, and M. Servel, "Implementing Asynchronous Transfer Mode Concepts: Main Results of the Prelude Experiment," *Globecom'87 Conf. Rec.*, pp. 1871-1875, Nov. 1987.

[4] T. Kozaki, Y. Sakurai, O. Matsubara, M. Mizukami, M. Uchida, Y. Sato, and K. Asano, "32x32 Shared Buffer Type ATM Switch VLSIs For B-ISDN," *ICC'91 Conf. Rec.*, pp. 711-715, June 1991.

[5] T.-C. Hou and D. M. Lucantoni, "Buffer Sizing for Synchronous Self-Routing Broadband Packet Switches with Bursty Traffic," *International Jour. of Digital & Analog Cabled Systems*, vol. 2, no. 4, pp. 253-260, Oct.-Dec. 1989.

[6] M. Hirano and N. Watanabe, "Characteristics of a Cell Multiplexer for Bursty ATM Traffic," *ICC'89 Conf. Rec.*, pp. 399-403, June 1989.

[7] L. Dittmann and S. B. Jacobsen, "Statistical Multiplexing of Identical Bursty Sources in an ATM Network," *Globecom'88 Conf. Rec.*, pp. 1293-1297, Nov. 1988.

[8] K. Y. Eng, M. J. Karol, and Y. S. Yeh, "A Growable Packet (ATM) Switch Architecture: Design Principles and Applications," *IEEE Trans. Comm.*, Vol. 40, No. 2, February 1992, pp. 423-430.

[9] K. Y. Eng and M. J. Karol, "The Growable Switch Architecture: A Self-Routing Implementation For Large ATM Applications," *ICC '91 Conference Record*, June 23-26, 1991, pp. 32.3.1-32.3.7.

[10] K. Y. Eng, et al., "A High-Performance Prototype 2.5 Gb/s ATM Switch For Broadband Applications," *GLOBECOM'92 Conf. Rec.*, pp. 111-117, Dec. 1992.

[11] K. Y. Eng, et al., "A Prototype Growable 2.5 Gb/s ATM Switch For Broadband Applications," *Jour. High Speed Networks*, vol. 1, no. 3, pp. 237-253, 1992.

[12] M. J. Karol and K. Y. Eng, "Performance of Hierarchical Multiplexing in ATM Switch Designs," *Supercomm/ICC '92 Conference Record*, June 1992.

[13] M. J. Karol, "Queueing in Optical Packet Switches," *SPIE Vol. 1787: Multigigabit Fiber Communications (1992)*, pp. 192-203, Sept. 1992.

[14] R. A. Thompson and P. P. Giordano, "Experimental Photonic Time-Slot Interchanger Using Optical Fibers As Reentrant Delay-Line Memories," *OFC'86 Tech. Digest*, p. 26, Feb. 1986.

[15] P. R. Prucnal, M. A. Santoro, and S. K. Sehgal, "Ultrafast All-Optical Synchronous Multiple Access Fiber Networks," *IEEE Jour. Select. Areas Commun.*, vol. SAC-4, no. 9, pp. 1484-1493, Dec. 1986.

[16] I. Chlamtac and A. Fumagalli, "QUADRO-Stars: High Performance Optical WDM Star Networks," *Globecom'91 Conf. Rec.*, pp. 1224-1229, Dec.'1991.

[17] Z. Haas, "The Staggering Switch: An Almost-All Optical Packet Switch," *OFC'92 Tech. Digest*, p. 133, Feb. 1992.

[18] Z. Haas, "On the Design of the Staggering Switch," *LEOS 1992 Summer Topical Meeting Digest on Optical Multiple Access Networks*, pp. 17-18, August 3-5, 1992.

[19] Z. Haas, "Gbps Optical Connectivity with the Staggering Switch," *SPIE Vol. 1787: Multigigabit Fiber Communications (1992)*, pp. 180-191, Sept. 1992.

[20] M. J. Karol, "Shared-Memory Optical Packet (ATM) Switch," *SPIE Vol. 2024: Multigigabit Fiber Communication Systems (1993)*, July 1993.

[21] A. Huang and S. Knauer, "Starlite: A Wideband Digital Switch," *GLOBECOM'84 Conf. Rec.*, pp. 121-125, Nov. 1984.

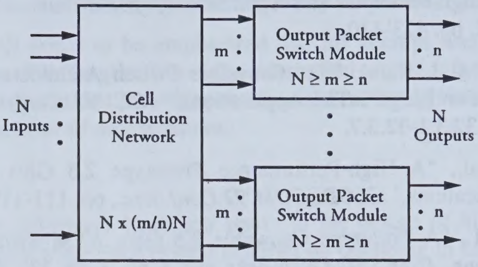

Growable Switch Architecture
N
Inputs
Cell
Distribution
Network
N x (m/n)N
m
m
Output Packet
Switch Module
N ≥ m ≥ n
Output Packet
Switch Module
N ≥ m ≥ n
n
n
N
Outputs

Fig. 1

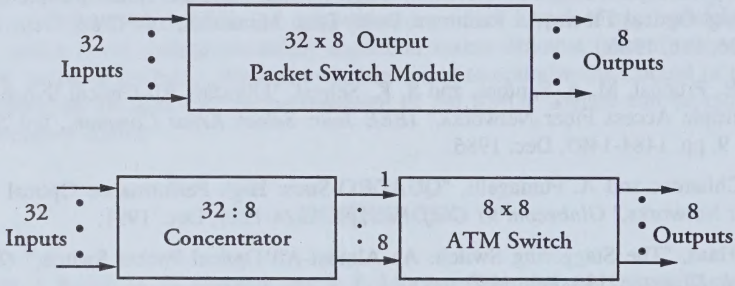

Concentrator-Based Output Packet Switch Module
32
Inputs
32 x 8 Output
Packet Switch Module
8
Outputs
32
Inputs
32 : 8
Concentrator
1
8
8 x 8
ATM Switch
8
Outputs

Fig. 2

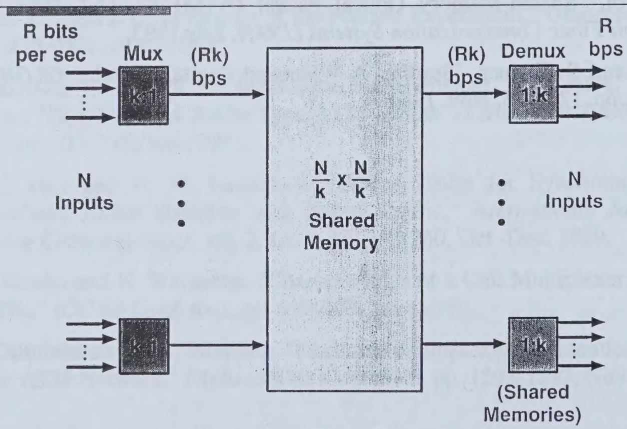

Hierarchical Multiplexing/Demultiplexing
R bits
per sec
Mux
k:1
(Rk)
bps
N
Inputs
k:1
N/k x N/k
Shared
Memory
(Rk)
bps
Demux
1:k
R
bps
N
Inputs
1:k
(Shared
Memories)

Fig. 3

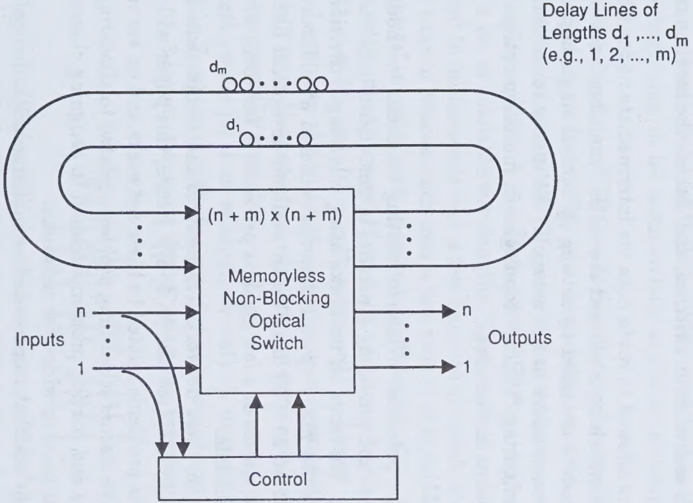

Delay Lines of
Lengths d_1 ,..., d_m
(e.g., 1, 2, ..., m)
d_m
d_1
(n + m) x (n + m)
Memoryless
Non-Blocking
Optical
Switch
Inputs
n
1
Outputs
n
1
Control

Fig. 4 Block Diagram of a Shared-Memory Optical Packet (SMOP) Switch

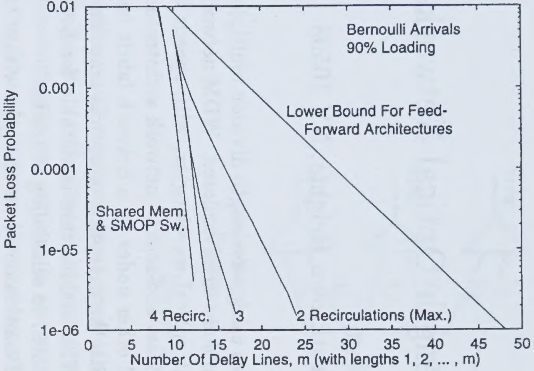


Fig. 5 Packet Loss Probability For n = 8 Inputs (And Outputs)

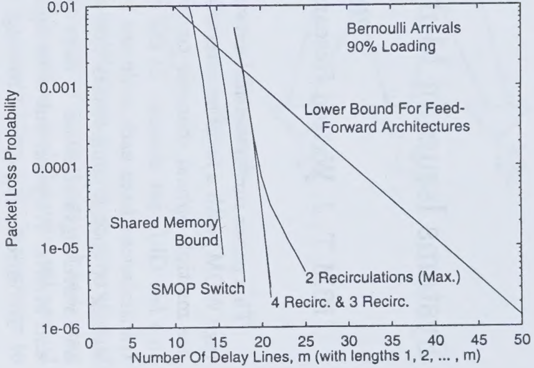


Fig. 6 Packet Loss Probability For n = 16 Inputs (And Outputs)

Systems Issues in Multi-Wavelength Optical Networks *

Rajiv Ramaswami
IBM T. J. Watson Research Center, Yorktown Heights, NY 10598

Abstract

This paper surveys optical network architectures using wavelength division multiplexing (WDM) with a focus on interesting systems problems to be explored. WDM networks use multiple optical channels centered at different optical wavelengths, with speeds up to a few Gb/s per channel. In the simplest "broadcast-and-select" network architecture, transmissions from each node are broadcast to all other nodes by a network fabric consisting of optical components that are passive (with the exception of optical amplifiers) and wavelength selection is done at the nodes. This architecture is suitable for LANs and MANs. We point out some open problems related to scheduling of traffic.

By introducing optical routing, switching, and conversion of wavelengths within the network, we can obtain significant flexibility in network design as well as reduce the total number of channels required. These architectures are known as "wavelength routing" architectures. We survey some recent work dealing with bounds on the number of channels required in networks with and without switches, and discuss the routing and wavelength assignment problem.

Networks where we are allowed to route data via intermediate nodes where electronic conversion and electronic switching is allowed are called "multihop" networks. A multihop "logical" topology can be obtained by setting up optical single-hop connections in a prescribed manner between nodes in the network. We survey some work done on the problem of optimally configuring "logical" topologies in multihop optical networks and mention some open problems in this area.

1 Introduction

Optical networks offer the potential of interconnecting hundreds to thousands of users covering local to wide areas and providing capacities of the order of gigabits per second to each user in the network. We focus on networks using wavelength division multiplexing (WDM). WDM networks use many optical channels centered at different wavelengths, thus making use of the vast 25 THz bandwidth available in optical fiber. With each channel being operated at speeds of a few gigabits per second, networks with terabit-per-second capacities are realizable.

Several survey papers on these network architectures and components have appeared in the literature recently [1, 2, 3] (see also [4, 5, 6]). Hence, this paper will focus more on posing interesting systems problems related to these networks and on surveying material not previously covered. We discuss scheduling problems related to allocating the available channels among the users and routing problems related to assigning channels and routes among source-destination pairs in wide-area networks.

Section 2 describes devices that can be used as building blocks for optical networks. Section 3 discusses broadcast-and-select networks, surveys work done on scheduling trans-

---

*This work was supported in part by grant no. MDA 972-92-C-0075 from ARPA.

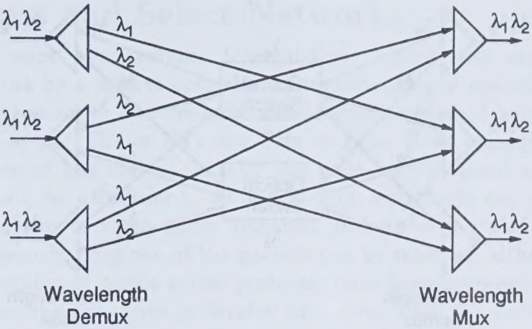


Figure 1: A wavelength router.

missions and mentions some open scheduling problems. Section 4 gives an overview of wavelength routing networks and discusses problems related to routing and assigning wavelengths to connections in the network. Section 5 considers multihop non-all-optical networks and discusses the problem of reconfiguring the network topology in response to changes in traffic patterns.

2 Components

Many of the components useful for building WDM networks are actively being researched today and are beginning to be commercially available, although they are still very expensive.

A *fixed-tuned* transmitter (FT) is a device that emits signals always at a fixed wavelength. A *fixed-tuned* receiver (FR) is a device that receives signals always at a fixed wavelength. A *tunable* transmitter (TT) is a device that can emit signals on any one of a set of available wavelengths. A *tunable* receiver (TR) is a device that can receive signals on any one of a set of available wavelengths. Transmitters and receivers that can be mechanically tuned in milliseconds over a few hundred channels are commercially available, and ones that tune in nanoseconds over a few tens of channels have been reported [7, 8].

A 2 × 2 optical coupler with coupling coefficient $0 \leq \alpha \leq 1$ is a two-input, two-output device with input powers P_1^i and P_2^i and output powers P_1^o and P_2^o given by $P_1^o = \alpha P_1^i + (1-\alpha)P_2^i$, and $P_2^o = \alpha P_2^i + (1-\alpha)P_1^i$. Larger couplers can be made by cascading 2 × 2 couplers. An $n \times n$ star coupler is a device with n inputs and n outputs that distributes $1/n$ of the power on each input port to each output port. 32 × 32 star couplers are commercially available.

A *wavelength router* [9] is an n–input, m–output, device that can be represented by an $n \times m$ matrix S, whose $(i,j)^{th}$ element, S_{ij} is the set of wavelengths connecting input i to output j. A 3-input, 3-output, device using two wavelengths λ_1, λ_2 represented by the matrix

$$S = \begin{bmatrix} \{\} & \{\lambda_1\} & \{\lambda_2\} \\ \{\lambda_2\} & \{\} & \{\lambda_1\} \\ \{\lambda_1\} & \{\lambda_2\} & \{\} \end{bmatrix}$$

is shown in Figure 1. These devices can be constructed using optical gratings and/or couplers and are entirely passive. A 11-input, 11-output router with 11 wavelengths per port has been demonstrated in integrated optics [10].

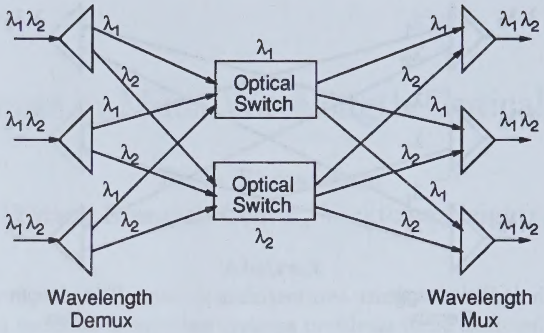


Figure 2: A generalized switch realized using simple switches.

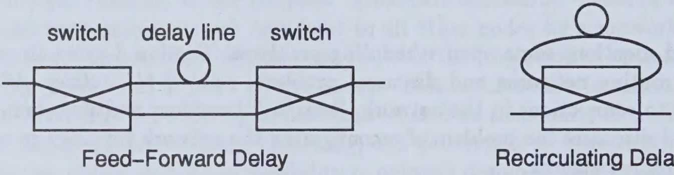


Figure 3: Two types of optical delays realized using delay lines and optical switches.

An optical 2 × 2 switch is similar to an electronic 2 × 2 switch. 2 × 2 fiber switches with low loss but slow switching times (milliseconds), as well as small-dimension (up to 8 × 8) lithium niobate switches with nanosecond switching times are commercially available. These are *simple* switches (or wavelength-flat switches) in that the switch does not distinguish between different wavelengths. All wavelengths coming in at a port are switched to the same output port. A 2 × 2 simple switch is essentially a variable optical coupler where $\alpha = 0$ or $\alpha = 1$. Some switches allow a continuous variation in α from 0 to 1. A *generalized* switch (or wavelength-selective switch) is capable of switching each wavelength at its input ports independent of the other wavelengths and is functionally equivalent to w simple switches, where w is the number of wavelengths (see Figure 2). For example, an acoustooptic filter functions as a 2 × 2 generalized switch [11]. We can think of the generalized switch as a reconfigurable wavelength router, i.e., as a device where the input-output wavelength assignments can be changed dynamically.

A *fixed wavelength converter* is a single-input, single-output device that converts data at a fixed input wavelength onto a fixed output wavelength. It is functionally equivalent to an FR followed by an FT. A *dynamic wavelength converter* is a single-input, single-output device that converts a single data stream at any input wavelength onto any output wavelength. It is functionally equivalent to a TR followed by a TT. Such devices have been demonstrated experimentally [12, 13].

Optical buffers are harder to realize than their electronic equivalents. Two basic forms of buffers using delay lines and optical switches are possible. The *feedforward* delay shown in Figure 3 can delay data by a fixed time unit. The *recirculating* delay shown in Figure 3 can store data for an indefinite amount of time in principle, but the realizable delay is limited in practice due to attenuation losses. Some interesting time-domain architectures using optical delay lines are explored in [14, 15, 16, 17] but we will not be considering them in this paper, since our focus is on WDM networks.

3 Broadcast and Select Networks

In broadcast and select architectures, transmissions from a node are broadcast to all nodes in the network by a passive optical medium consisting of optical couplers, usually arranged in a star topology. The available wavelengths are shared by the nodes in some fashion. Nodes may use TTs or FTs and TRs or FRs. If we consider a network with TTs and TRs, then at any instant, exactly one node may transmit on a given channel (otherwise there will be a collision). At any instant, each node can receive on exactly one channel, so if two or more nodes transmit packets to a given node on different channels simultaneously, only one of the packets can be received, although there will be no collisions. A number of media-access protocols have been proposed and are surveyed in [1, 2]. Most assume that time is divided into slots, that the nodes in the network are synchronized to a common slot clock, and that the tuning times are very small. Synchronization can be achieved in principle if each node measures its propagation delay from/to a common point in the network and adjusts its clock relative to a common reference clock [18]. In practice, it is also important to minimize the bit clock acquisition time after tuning has been completed.

3.1 Scheduling

The static scheduling problem for a network with n nodes and w channels, with each node having a T and a TR, may be stated as follows. Given a matrix indicating traffic between each source and destination node in units of slots, determine a schedule of minimum number of slots subject to the following constraints: (1)in any given slot, at most one node can be scheduled to transmit on a given channel, (2) in any given slot, a node can be scheduled to receive on at most one channel, and (3) it takes a transmitter or receiver τ slots to tune from one channel to another. Similar constraints can be used for networks with FTs, TRs and TTs, FRs. The case $\tau = 0, w \geq n$ is identical to the minimum length time slot assignment (TSA) problem studied in satellite-switched time division multiple access (SS/TDMA) systems for which optimal algorithms are known (see [19, 20] and the references therein).

In dynamic scheduling, the traffic matrix is updated during each time slot to reflect new arrivals and departures. Hajek and Weller [21, 22] studied the dynamic scheduling problem for the case $\tau = 0, w \geq n$ for arbitrary traffic matrices where the traffic, represented by two parameters α, S, satisfies the constraint that over any time period of S slots, not more than αS packets arrive at a source and not more than αS packets bound for a single destination arrive at all the sources. The access delay D is the waiting time of the packet (in slots) before it is successfully transmitted by the source. The maximum line backlog L is the maximum of the number of packets queued at any source, and the total number of packets queued at all sources for a single destination. Let d_{prop} denote the one-way propagation delay between two nodes (assumed to be the same for all pairs of nodes). With $d_{prop} = 0$, they showed that the maximum and average access delay $\overline{D}$ and D_{max} respectively, and L, are $O(\alpha S)$ for some simple scheduling algorithms, if α is constrained to be below a certain constant value (depending on the algorithm used). We can also show easily that $\Omega(\alpha S)$ is a lower bound for all these quantities. When d_{prop} is large, they give an algorithm that has $D_{max} \leq 2S$ and $L \leq 2\alpha S$, provided α is $O(1/\sqrt{n})$ and $n \leq (\alpha S)^{2-\epsilon}$ for any $\epsilon > 0$. They also give a randomized algorithm with $\overline{D} \leq \epsilon d_{prop}$, and $D_{max} \leq (1+\epsilon)d_{prop}$, for $\alpha \leq 1/4$ and $\epsilon > 0$.

In practice, the tuning time τ can be significant (a few slots). The static scheduling problem for this case ($\tau =$ constant) has been considered by Pieris and Sasaki [23] and Aggarwal et al. [24] when each node is assumed to send one packet to every node. For a

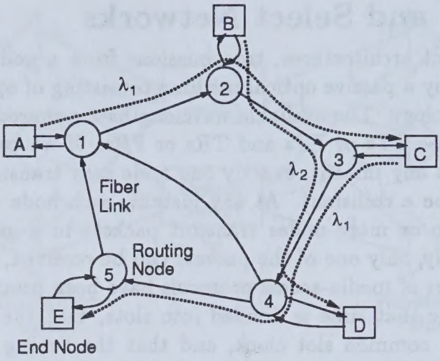


Figure 4: A WDM network consisting of routing nodes interconnected by point-to-point fiber-optic links. The routing nodes have end-nodes attached to them that form the sources and destinations for network traffic.

network with n nodes, it is shown that any schedule requires $\Omega(n\sqrt{\tau})$ slots and schedules using $O(n\sqrt{\tau})$ slots are given. Arbitrary traffic matrices are yet to be considered.

Another case is when the tuning time is proportional to the number of channels through which a transmitter or receiver must tune. This has not yet been considered.

The scheduling problem for the case where the underlying topology is not a broadcast topology (as in the wavelength routing networks to be considered next) has also not been studied.

4 Wavelength Routing Networks

The simple broadcast-and-select architecture is suitable only for a small number of nodes (a few hundred) in local to metropolitan areas. As the number of nodes increases, the energy loss due to splitting of transmitted power among all the nodes increases. Moreover, since the pool of available wavelengths is limited and cannot be spatially reused, system capacity suffers. The solution to these problems is to use components in the network that are capable of routing, switching, and converting wavelengths. An example of such a network is shown in Figure 4. Each link in the network is capable of supporting a certain number of wavelengths. Each routing node contains wavelength routers or generalized switches and/or wavelength converters and is capable of routing a wavelength coming in to an input port to any of its output port independent of the other wavelengths. However two signals coming in on the same wavelength at two input ports cannot be routed to the same output port. If wavelength converters are present, then the routing node is also capable of taking a signal coming in on a given wavelength and sending it out at a different wavelength on one of the output ports. The end nodes in the network serve as sources and sinks of traffic at any of the available wavelengths. These networks will most likely be used to provide circuit-switched unicast or multicast connections between end-nodes.

4.1 Non-Adaptive Networks

In *non-adaptive* or *non-reconfigurable* networks, the routing pattern at the routing nodes is fixed at the time of network design and cannot be changed. These networks can be built entirely out of passive optical components (excepting perhaps optical amplifiers). The only freedom remaining after the network is designed is in the choice of wavelengths at which each end-node transmits and receives. We assume that each end-node has a tunable transmitter (TT) and a tunable receiver (TR), unless otherwise stated. The network may use wavelength routers, couplers, and fixed wavelength converters, but not switches or dynamic wavelength converters. The simplest example of a non-adaptive network is the broadcast-and-select network discussed earlier, where n wavelengths are required to support permutation routing in a network with n nodes. (A permutation network is a network that can successfully route all sets of connections where each transmitter is to be connected to a single receiver and each receiver to a single transmitter.)

A simple model for this problem is to represent the network by a bipartite multigraph. (An equivalent model is the matrix representation used earlier to characterize routers.) The n nodes in one set are the transmitters and the n nodes in the other set are the receivers. Each edge between a transmitter i and receiver j is labelled with two wavelengths $\lambda^t_{ij}, \lambda^r_{ij}$. This indicates that if transmitter i transmits using wavelength λ^t_{ij}, the transmission will reach receiver j at λ^r_{ij}. If no wavelength converters are used, then $\lambda^t_{ij} = \lambda^r_{ij}$. Note that there can be multiple edges between a transmitter-receiver pair labelled with different wavelengths indicating that there are many possible wavelength pairs using which the two can communicate. For each permutation (or traffic state) where transmitter t is connected to receiver $\pi(t)$, we must assign a wavelength λ^t to t and a wavelength $\lambda^{\pi(t)}$ to $\pi(t)$. The problem is to determine the edges, find a labelling for them, and an assignment of wavelengths for each permutation such that (1) there exists an edge labelled $\lambda^t, \lambda^{\pi(t)}$ between t and $\pi(t)$ (so that the wavelength assignment to t and $\pi(t)$ is satisfied), and (2) for all transmitters $t' \neq t$ such that there exists an edge between t' and $\pi(t)$ labelled $\lambda^*, \lambda^{\pi(t)}$, t' is not assigned wavelength λ^* (since if t' transmits at wavelength λ^*, there will be collision at $\pi(t)$).

Barry and Humblet [25] have shown that $\Omega(\sqrt{n})$ wavelengths are required to support permutations. This bound is obtained essentially by observing that the number of possible "network" states (w^{2n}, assuming w wavelengths are available) must be larger than the number of "traffic" states ($n!$ for permutations). The existence of a wide-sense non-blocking permutation network with $O(\sqrt{n \log n})$ wavelengths is shown in [24, 26] without using wavelength converters and in [27] using wavelength converters. The construction of such a network without wavelength converters using $O(\sqrt{n}2^{(\log n)^{0.8+o(1)}})$ wavelengths is given in [24].

All the results above assume that it is possible to accommodate the desired interconnection pattern of wavelengths specified by the bipartite multigraph (or the matrix) by using a suitable topology. A more important problem that has not yet been considered is how to determine the wavelength assignment when we are given the topology representing the interconnection of routing nodes. In this case, we are allowed to choose the configuration of each routing node but not the links by which these nodes are interconnected.

4.2 Adaptive Networks

In *adaptive* or *reconfigurable* networks, the routing pattern at the routing nodes can be changed dynamically, at least on a circuit-by-circuit basis, if not on a packet-by-packet basis by using optical switches and/or dynamic wavelength converters. This allows us to have significantly more "network states" thus reducing the number of wavelengths required at the cost of increasing the complexity of control.

When we are allowed to choose the topology, tight bounds of $w = \Theta(\frac{n \log n}{k})$ are shown for permutation routing in rearrangeably non-blocking adaptive networks [28, 24] and wide-sense non-blocking adaptive networks [24], without wavelength converters. Here, w

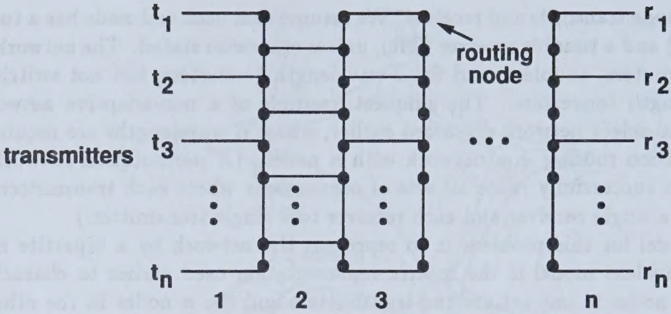


Figure 5: An example to illustrate the difference between networks with and without wavelength converters.

is the number of wavelengths, k is the number of 2×2 generalized switches and n is the number of nodes.

For networks with simple switches, Barry and Humblet [25] show that $\Omega(n \log \frac{n}{w^2})$ switches are required to support permutation routing and Aggarwal et al. [24] show the existence of wide-sense non-blocking permutation networks using $O(n \log \frac{n \log w}{w^2})$ switches and a construction using $O(n \log \frac{n 2^{(\log n)^{0.8+o(1)}}}{w^2})$ switches. When each node has an FT and a TR, the number of simple switches required is $\Theta(n \log \frac{n}{w})$ [29].

4.2.1 Routing and Wavelength Assignment in Arbitrary Networks

We now focus on the problem of routing and assigning wavelengths to connections in networks where the topology is already given. First consider the case where wavelength converters are not used. In this case, each connection must be assigned a wavelength and path in the network subject to the constraint that two connections using the same wavelength must be routed on edge-disjoint paths. In Figure 4, a connection from node A to node C is carried on wavelength λ_1, another connection from node C to node E also is carried on λ_1, but the connection from node B to node D must be carried on a different wavelength λ_2.

Now consider the case where dynamic wavelength converters can be used wherever necessary. In this case, each connection must be assigned a path in the network and a wavelength on each edge in the path subject to the constraint that not more than Λ connections can use an edge, where Λ is the number of wavelengths available on the edge. This is identical to the routing problem studied in circuit-switched telephone networks[30, 31].

The difference between these two cases can be significant as shown in the following example taken from [24]. The network shown in Figure 5 has n transmitters and n receivers with a total of $O(n)$ routing nodes and $O(n^2)$ edges. Suppose that we are interested in establishing a connection between transmitter i and receiver $n - i + 1$. For any choice of n paths satisfying these connections, any two paths will intersect on at least one edge. Thus for the case where no wavelength converters are available, each path must be assigned a different wavelength and hence n wavelengths are required. It is possible to construct n paths so that not more than two of them will share a single edge. This means that a network using wavelength converters will require only two wavelengths to satisfy this traffic demand. An interesting intermediate situation when the number of wavelength converters available is limited is considered in [32].

The problem of optimally choosing routes and wavelengths on the routes for arbitrary networks appears to be a hard problem. Several heuristics have been proposed for adaptive networks with and without wavelength converters [33, 34, 32] and their performance for some example networks have been obtained via simulation. In [35], the problem of maximizing carried traffic is posed as an integer linear programming problem and an upper bound is obtained by relaxing the integrality constraints and transforming the problem to a linear programming problem, which can then be solved easily. This bound is tight when a large number of wavelengths is available. While the bound is easy to compute for networks with wavelength converters (telephone networks), for networks without wavelength converters, the number of variables in the linear program can be exponential in the number of source-destination pairs in the network. Bounds that can be computed more easily, as well as good bounds for the case where the number of wavelengths is small will be useful. Heuristics whose performance can be better quantified are highly desirable.

5 Non All-Optical Networks

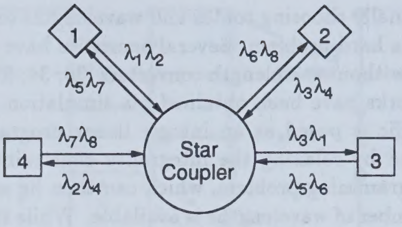
So far, we have focussed on *all-optical networks*, i.e., networks where data is carried from the source to its destination optically without any conversions to electronic form and reconversion back to optical form in between. The main advantage of all-optical networks is that they are transparent to bit rate, modulation schemes, and protocols, once an optical connection is set up. However they are mainly useful for providing a large number of circuit-switched connections. A network where data is electronically switched and routed through intermediate nodes is called a *multihop network*. In a multihop network, we set up logical connections, or circuits using all-optical paths between pairs of nodes. Using this *logical topology*, a packet can be routed through multiple intermediate nodes (and multiple logical connections) before reaching its final destination. Each intermediate node receives the packet, converts it into electronics, processes it and sends it out on another logical connection. This allows us to do packet switching (albeit using electronic switches) and is also useful if we do not have enough wavelengths/optical switches to satisfy the traffic demands all-optically. An example of imposing a logical topology over an underlying broadcast topology is shown in Figure 6.

5.1 Reconfiguration

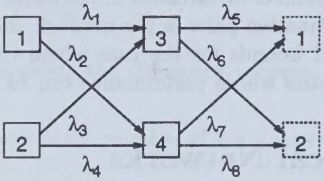
A major advantage offered by multihop optical networks is that the logical topology can be changed by simply retuning transmitters and/or receivers, and/or changing the switch settings inside the network. Thus it is possible to optimize the logical topology for a given traffic pattern. Metrics considered typically are the minimization of the maximum congestion on a logical link (defined as the ratio of the traffic flow to the capacity on that link) or the mean packet delay (usually only the mean propagation delay, or equivalently the average number of hops between source-destination pairs, assuming that queueing delays are negligible compared to propagation delays, which is true at light loads).

For uniform traffic, lower bounds on the average number of hops and the average congestion (which in turn is a lower bound on the maximum congestion) for any graph with a given maximum nodal degree are derived in [36]. Several regular topologies such as shufflenets and de Bruijn graphs and corresponding routing schemes are good candidate topologies [37].

Lower bounds on the desired metrics for non-uniform traffic are yet to be derived. For this case, several heuristics to optimize the logical topology have been proposed and are surveyed in [3]. Only simulation results for some simple networks are available



(a) Star Physical Network



(b) Shufflenet Logical Topology

Figure 6: An example to illustrate the multihop approach. Each node is provided with two FTs and FRs. The logical topology obtained is determined by the wavelengths assigned to the FTs and FRs and is also shown.

to evaluate their performance. These algorithms mostly work on broadcast underlying topologies and need to be extended to work over arbitrary underlying topologies, such as the wavelength-routing topologies discussed earlier. However very little work has been done on the problem of how to move from one logical topology to another as the traffic pattern changes.

One approach suggested in [38] is to evolve the logical topology "on the fly" as new connection requests arrive but not break any existing logical link that has already been set up. The idea is to route each connection on a single-hop path if possible, setting up such a link if it does not already exist. If it is not possible to set up such a link, then the connection is routed through a path with the minimum number of hops possible. However, since links once set up cannot be taken down, the topology evolves eventually to a fixed topology that cannot be changed if the traffic pattern changes, since no links can be set up. There is also the danger that this topology may not be *connected*, which is the only case when some links are allowed to be taken down and other links established to make the topology connected.

In general, when a link is taken down, packets are lost and must be recovered. There is also an associated control overhead with setting up or taking down a logical link. Thus there is a cost for moving from one topology to another (e.g. proportional to the number of edges that are different in the new topology compared to the old one). There is however, also a cost for staying in a non-optimal topology when the traffic pattern has changed (e.g., proportional to the average number of hops). These two measures need to be traded off against each other in determining the optimal reconfiguration strategy.

One approach [39] attempts to minimize the cost of reconfiguration by evolving the initial topology to the desired final one through a series of intermediate topologies using branch exchange operations, where an intermediate topology is obtained from its predecessor by swapping exactly two edges.

Acknowledgements: I would like to thank Kumar Sivarajan for his many comments.

References

[1] R. Ramaswami, "Multi-wavelength lightwave networks for computer communication," *IEEE Communications Magazine*, vol. 31, pp. 78–88, Feb. 1993.

[2] B. Mukherjee, "WDM-based local lightwave networks-part I: Single-hop systems," *IEEE Network*, vol. 6, pp. 12–27, May 1992.

[3] B. Mukherjee, "WDM-based local lightwave networks-part II: Multihop systems," *IEEE Network*, vol. 6, pp. 20–32, July 1992.

[4] P. E. Green, *Fiber-Optic Networks*. Prentice Hall, 1992.

[5] M. J. Karol, G. Hill, C. Lin, and K. Nosu, eds., *IEEE/OSA J. Lightwave Tech.: Special Issue on Broadband Optical Networks*, May/June 1993.

[6] N. K. Cheung, G. Nosu, and G. Winzer, eds., *IEEE JSAC: Special issue on Dense WDM Networks*, vol. 8, Aug. 1990.

[7] R. C. Alferness, U. Koren, L. L. Buhl, B. I. Miller, M. G. Young, T. L. Koch, G. Raybon, and C. A. Burrus, "Widely tunable InGaAsP/InP laser based on a vertical coupler filter with 57-nm tuning range," *Appl. Phys. Lett.*, vol. 60, pp. 3209–3211, 1992.

[8] V. Jayaraman, D. A. Cohen, and L. A. Coldren, "Extended tuning range in a distributed feedback InGaAsP laser with sampled gratings," in *OFC'92 Tech. Digest*, no. WL12, p. 165, Feb. 1992.

[9] R. A. Barry and P. A. Humblet, "Latin routers: Design and implementation," *IEEE/OSA J. Lightwave Tech., to appear*, 1993.

[10] C. Dragone, C. A. Edwards, and R. C. Kistler, "Integrated optics $N \times N$ multiplexer on silicon," *IEEE Photonics Tech. Lett.*, vol. 3, pp. 896–899, Oct. 1991.

[11] K.-W. Cheung, "Acoustooptic tunable filters in narrowband WDM networks: system issues and network applications," *IEEE JSAC.*, vol. 8, pp. 1015–1025, Aug. 1990.

[12] B. Glance, J. M. Wiesenfield, U. Koren, A. H. Gnauck, and H. M. Presby, "High-performance optical wavelength shifter," *Electron. Lett.*, vol. 28, pp. 1714–1715, 1992.

[13] S. Suzuki, M. Nishio, T. Numai, M. Fujiwara, M. Itoh, S. Murata, and N. Shimosaka, "A photonic wavelength-division switching system using tunable laser diode filters," *IEEE/OSA J. Lightwave Tech.*, vol. 8, pp. 660–665, May 1990.

[14] D. K. Hunter, P. E. Barnsley, I. Andronovic, and B. Culshaw, "Architectures for optical TDM switching," in *Proc. SPIE Multigigabit Fiber Commun. Conf.*, vol. 1787, pp. 204–214, 1992.

[15] H. F. Jordan, K. Y. Lee, and D. Lee, "Multichannel time slot permuters," in *Proc. SPIE High-Speed Fiber Networks and Channels Conf.*, vol. 1784, pp. 252–261, 1992.

[16] S. V. Ramanan, H. F. Jordan, and J. R. Sauer, "A new time domain, multistage permutation algorithm," *IEEE Trans. on Inform. Theory*, vol. 50, pp. 171–173, Jan. 1990.

[17] I. Chlamtac and A. Fumagalli, "An optical switch architecture for Manhattan networks," *IEEE JSAC*, vol. 11, pp. 550–559, May 1993.

[18] G. Semaan and P. A. Humblet, "Timing and dispersion in WDM optical star networks," in *Proc. Infocom'93*, pp. 573–577, 1993.

[19] S. Chalasani and A. Varma, "Efficient time slot assignment algorithms for SS/TDMA systems with variable bandwidth beams," in *Proc. Infocom'91*, pp. 658–667, 1991.

[20] I. S. Gopal, G. Bongiovanni, M. A. Bonuccelli, D. T. Tang, and C. K. Wong, "An optimal switching algorithm for multibeam satellite systems with variable bandwidth," *IEEE Trans. on Commun.*, vol. 40, pp. 2475–2481, 1982.

[21] T. Weller and B. Hajek, "Scheduling non-uniform traffic in a packet switching system with small propagation delay," *Submitted to Infocom'94*, 1993.

[22] B. Hajek and T. Weller, "Scheduling non-uniform traffic in a packet switching system with large propagation delay," *Submitted to IEEE Trans. on Inform Theory*, 1993.

[23] G. R. Pieris and G. H. Sasaki, "Scheduling transmissions in WDM broadcast and select networks," *Submitted to IEEE/ACM Trans. on Networking*, 1993.

[24] A. Aggarwal, A. Bar-Noy, D. Coppersmith, R. Ramaswami, B. Schieber, and M. Sudan, "Efficient routing and scheduling algorithms for optical networks," Tech. Rep. RC 18967, IBM Research Report, June 1993.

[25] R. A. Barry and P. A. Humblet, "Bounds on the number of wavelengths needed in WDM networks," in *LEOS'92 Summer Topical Mtg. Digest*, pp. 21–22, 1992.

[26] R. A. Barry, "An all-optical non-blocking $M \times M$ switchless connector with $O(\sqrt{M \log M})$ wavelengths and without wavelength changers," *Electron. Lett.*, vol. 29, pp. 1252–1254, July 1993.

[27] R. A. Barry and P. A. Humblet, "On the number of wavelengths and switches in all-optical networks," *IEEE Trans. on Commun., to appear*, 1993.

[28] J. Sharony, T. E. Stern, and Y. Li, "The universality of multi-dimensional switching networks," *Submitted to IEEE/ACM Trans. on Networking*, 1993.

[29] G. R. Pieris and G. H. Sasaki, "A linear lightwave Beneš network," *To appear in IEEE/ACM Trans. on Networking*, 1993.

[30] A. Girard, *Routing and dimensioning in circuit-switched networks.* Addison-Wesley, 1990.

[31] R. J. McEliece and K. N. Sivarajan, "Maximizing marginal revenue in generalized blocking service networks," in *Proc. 30th Annual Allerton Conference on Communications, Control and Computing*, vol. 18, pp. 455–464, 1992.

[32] K.-C. Lee and V. O. K. Li, "Routing and switching in a wavelength convertible lightwave network," in *Proc. INFOCOM'93*, pp. 578–585, 1993.

[33] K. Bala, T. E. Stern, and K. Bala, "Algorithms for routing in a linear lightwave network," in *Proc. Infocom'91*, pp. 1–9, 1991.

[34] I. Chlamtac, A. Ganz, and G. Karmi, "Lightpath communications: an approach to high-bandwidth optical WAN's," *IEEE Trans. on Commun.*, vol. 40, pp. 1171–1182, July 1992.

[35] R. Ramaswami and K. N. Sivarajan, "Optimal routing and wavelength assignment in all-optical networks," Tech. Rep. RC 18964, IBM Research Report, June 1993.

[36] K. N. Sivarajan and R. Ramaswami, "Lightwave networks based on de bruijn graphs," *Submitted to IEEE/ACM Trans. on Networking*, 1993.

[37] K. N. Sivarajan, "Multihop logical topologies in gigabit lightwave networks," *IEEE Lightwave Telecom. Syst. Magazine*, Aug. 1992.

[38] J. S. Auerbach and R. K. Pankaj, "Use of delegated tuning and forwarding in WDMA networks," *IBM Research Report RC 16964; also to appear in IEEE Trans. on Commun.*, 1991.

[39] J.-F. P. Labourdette, A. S. Acampora, and G. W. Hart, "Reconfiguration algorithms for rearrangeable lightwave networks," in *Proc. INFOCOM'92*, pp. 2205–2214, 1992.

Design of Robust Nonlinear Feedback Controls: Concepts, Tools, and Procedures *

R. A. Freeman P. V. Kokotovic

Center for Control Engineering and Computation
Department of Electrical & Computer Engineering
University of California, Santa Barbara, CA 93106

ABSTRACT

We review some of the existing results on the Lyapunov design of robustly stabilizing feedback laws for uncertain nonlinear systems. Using the concept of a robust control Lyapunov function (rclf), we present robust and adaptive backstepping tools and demonstrate how they can be used in systematic design procedures.

1 INTRODUCTION

In the last decade, major progress has been made in the development of a geometric theory of nonlinear feedback systems (as summarized in [1, 2]). Combined with the well-established Lyapunov stability theory, these geometric results form a solid foundation on which to build systematic design procedures. To become practical, such procedures must overcome two important obstacles. First, they must expand the geometric methods to incorporate uncertainties in the system models. Second, they must deal with the crucial shortcoming of the Lyapunov approach, namely, the lack of tools for the systematic construction of Lyapunov functions. In this paper we survey recent advances in the latter direction. We present some basic tools and illustrate how they can be combined in recursive design procedures. The references in this paper will be presented primarily at the end of each subsection; they are intended to be representative rather than exhaustive.

2 CONCEPTS

Concepts from Lyapunov stability theory are suitable for analysis, but how can they be converted into design tools? One possibility is to explicitly express the dependence of the derivative $\dot{V}(x,u)$ of a candidate Lyapunov function $V(x)$ on the control variable u, and then show that $\dot{V}$ can always be made negative by a judicious choice of u. This leads to the definition of the *control Lyapunov function* (clf) and its robust counterpart, the *robust control Lyapunov function* (rclf). In this section we discuss the relationship between the existence of such functions and the stabilizability of nonlinear systems. We also discuss the relationship between the rclf and an associated differential game; this gives us a recipe for choosing a feedback law which is optimal with respect to a reasonable performance criterion.

2.1 Stabilizability

We are interested in designing continuous state feedback laws $u = k(x)$ which drive the state x to the origin $x = 0$ from any initial condition. Our goal is to achieve global uniform asymptotic stability (GUAS) of the origin, or, when this is not possible, global uniform stability and ultimate boundedness (GUSUB) with respect to some positively

---

*This work was supported by the U.S. Department of Energy Grant DE-FG-02-88-ER-13939, by the Air Force Office of Scientific Research Grant AFOSR F49620-92-J-0004, and by a National Science Foundation Graduate Research Fellowship.

invariant residual set containing the origin. We define three types of stabilizability in the order of their desirability. We say a system is *globally asymptotically stabilizable* (GAS) when there exists a feedback law which renders the origin GUAS. We say a system is *globally practically stabilizable* (GPS) when for every $\varepsilon > 0$ there exists a feedback law which renders the system GUSUB with respect to a subset of an ε-ball around the origin. Finally, we say a system is *globally stabilizable* (GS) when there exists a feedback law which renders the system GUSUB. These three types of stabilizability have robust counterparts RGAS, RGPS, and RGS, all defined when stability or boundedness holds for any admissible uncertainty.

2.2 Control Lyapunov function (clf)

A *control Lyapunov function* (clf) for a nonlinear system $\dot{x} = f(x, u)$ is a C^1, positive definite, radially unbounded function $V(x)$ for which there exists a bounded set Π such that

$$x \notin \Pi \implies \exists u \text{ such that } \nabla V(x) \cdot f(x, u) < 0 \tag{1}$$

Thus V is a **clf** when for every $x \notin \Pi$ we can find some value of the control u which makes $\dot{V}$ negative. At first it would seem that finding a **clf** for the system is tantamount to solving the stabilization problem. However, several issues remain to be addressed once a **clf** is found. One of these issues is whether or not a *continuous* feedback law $u = k(x)$ can be found to satisfy (1). For this we must place additional restrictions on the system and on the **clf**. If we assume our system is *affine in the control*, then the existence of a **clf** implies that the system is GS. If in addition we have $\Pi = \{0\}$, then the existence of a **clf** implies that the system is GPS. In order to guarantee asymptotic (rather than practical) stabilizability, our **clf** must also satisfy a *small control property* (**scp**) which says that small values of u satisfy (1) whenever x is small. Thus for systems affine in the control, the existence of a **clf** (with $\Pi = \{0\}$) satisfying the **scp** is equivalent to GAS (necessity comes from converse Lyapunov theorems).

Once a **clf** is known, a stabilizing feedback law $u = k(x)$ can be found analytically or numerically. But how do we find a **clf** for a given system? For globally feedback linearizable systems, we can use the linearizing coordinate transformation and find a **clf** by solving the linear Lyapunov matrix equation. For broader classes of systems, we can find a **clf** using recursive procedures such as backstepping (discussed below). In general, however, the question of how to find a **clf** remains open.

The basic theorems relating stabilizability to the existence of **clf**'s are found in [3, 4, 5]. The connection between **clf**'s and asymptotic controllability is described in [6]. Some sufficient conditions for the existence of **clf**'s are given in [7].

2.3 Robust control Lyapunov function (rclf)

We now extend the **clf** concept to systems with uncertainties. For reasons mentioned above, we only consider systems which are affine in the control:

$$\dot{x} \;=\; f(x, u, w) \;=\; f_0(x, w) + \sum_{i=1}^{m} g_i(x, w)\, u_i \tag{2}$$

Here $u = (u_1, \ldots, u_m)$ is the control and w is a disturbance representing the system uncertainties. We assume the disturbance takes its values in some compact set W; in general we allow W to depend on x in a continuous manner. A *robust control Lyapunov function* (rclf) for the system (2) is a C^1, positive definite, radially unbounded function $V(x)$ for which there exists a bounded set Π such that

$$x \notin \Pi \implies \exists u \text{ such that } \max_{w \in W} [\nabla V(x) \cdot f(x, u, w)] < 0 \tag{3}$$

Thus V is an **rclf** for our system when for every $x \notin \Pi$ we can find some value of the control u which makes $\dot{V}$ negative regardless of the value of the disturbance $w \in W$. The basic theorem for **rclf**'s is similar to the one for **clf**'s: the existence of an **rclf** for the system (2) implies RGS; if in addition we have $\Pi = \{0\}$ then the system is RGPS; and if in addition the **scp** is satisfied then the system is RGAS. Note, however, that the **scp** is even more restrictive when uncertainties are present—it implies that the uncertainties vanish as x approaches the origin! Therefore, for systems with uncertainties we often deal with RGS or RGPS rather than RGAS. In the case where we have RGS but not RGPS, the shape and size of the achievable residual set depend on the **rclf**. By adjusting the **rclf** we can sometimes arbitrarily reduce the size of the residual set in some directions by allowing it to peak in other directions.

How do we find an **rclf** for a given system? One might start by finding a **clf** for the nominal system (e.g., $w = 0$) and then checking to see if the **clf** is also an **rclf**. It is obvious from (1) and (3) that this method works when w enters the system equations at exactly the same place as u; this is commonly known as the *matching condition.* This method also works if the mismatched uncertainties are sufficiently small. In general, however, we cannot obtain an **rclf** from just any **clf** for the nominal system. We will show in Section 3 how backstepping can be used to find **rclf**'s for systems with large mismatched uncertainties.

The basic theorems relating robust stabilizability to the existence of **rclf**'s have been recently proven in [8]. The method of finding an **rclf** for a system satisfying the matching condition by using any **clf** for the nominal system can be found for example in [9, 10]. The analysis showing that this method also works when mismatched uncertainties are sufficiently small can be found in [11]. A more general matching condition under which one can use any **clf** for the nominal system is given in [12].

2.4 Performance and optimality

Our design is not over once we find an **rclf**; we are still faced with the choice of an appropriate feedback law. Usually at least one feedback law is known (perhaps obtained from the construction of the **rclf**), but there is no guarantee that it will provide adequate performance. We would prefer to choose a feedback law according to some reasonable optimality criterion, but how do we achieve optimality while maintaining the negativity of $\dot{V}$? It turns out that every **rclf** is an optimal upper value function of an associated differential game with a reasonable cost functional. We can therefore use this game to obtain an optimal feedback law which renders $\dot{V}$ negative as desired. The calculation of the optimal feedback law can be formulated as a static nonlinear programming problem involving V, the system equations and constraints, and adjustable design parameters. If we follow this procedure for choosing the feedback law, then the cost functional of the game gives us some measure of the performance of the closed-loop system. These results can be found in [8], and they represent a solution to the inverse optimal control problem (surveyed in [13]) in a game setting. Other authors have also used the construction of a cost functional as part of the design of a stabilizing feedback law; see for example [5] for the state feedback case and [14] for the output feedback case. Homogeneity properties of the system are exploited in [15] in the construction of a cost functional.

Instead of constructing a cost functional as a step in the design procedure, a more common framework is to directly pose the optimal control problem for a *given* cost functional. One such formulation leads to nonlinear H_∞ control as described for example in [16] and the references therein; these results are based on the game theory approach to linear H_∞ control as described in [17]. The nonlinear H_∞ framework requires the solution of a Hamilton-Jacobi type partial differential equation, the solvability of which may not be feasible other than locally. Consequently only local stability results are usually obtained in this framework.

After we find an **rclf** V for our system and choose some preferred feedback law based on V, we can implement the control and achieve stability as desired. However, we must still ask whether or not there is a better **rclf** for our system. We will see below that our choice for an **rclf** can be crucial for the system performance, and the question of finding a "good" **rclf** remains largely open. A case study on the effects of different choices of rclf's and feedback laws on the system performance can be found in [18].

3 TOOLS

We will use a variety of design tools to find a stabilizing feedback law for the following second-order single-input system:

$$\begin{aligned} \dot{x}_1 &= x_2 + w_1 x_1^3 && (4) \\ \dot{x}_2 &= u + w_2 && (5) \end{aligned}$$

The disturbance $w = (w_1, w_2)$ represents the uncertainty in the system, and our choice of a design tool depends on what we know about w. If we assume knowledge of a compact set W such that $w(t) \in W$ for all $t \geq 0$, then we can use the robust backstepping tool of Section 1. In Section 2 we modify the backstepping tool for the case where the set W exists but is *unknown.* If w is an unknown *constant,* we can use the adaptive backstepping tool of Section 3. Finally, in Section 4 we refer to a set of tools available for the construction of a *linear* feedback law.

3.1 Robust backstepping

In this section we assume $w(t) \in W$ for all $t \geq 0$, where $W = [-1,1]^2$. The design involves two major steps: first, we must find an **rclf** for the system; second, we must choose a feedback law according to the **rclf**.

We construct an **rclf** by first treating w_1 as a matched uncertainty: we pretend the state x_2 is the control and consider only the first state equation (4). The uncertainty w_1 is matched with respect to x_2, and so we can use a **clf** $V_1(x_1)$ for the "nominal" system $\dot{x}_1 = x_2$ as an **rclf** for the first state equation (4). An obvious choice for V_1 is $V_1(x_1) = \frac{1}{2}x_1^2$, and we obtain

$$\max_w \dot{V}_1 \;=\; \max_w \left(x_1 x_2 + w_1 x_1^4\right) \;=\; x_1 x_2 + x_1^4 \qquad (6)$$

which can clearly be made negative (for x_1 nonzero) by choice of x_2. We next need to find a smooth feedback law $k_1(x_1)$ for x_2 such that (6) is negative definite. A simple choice is $x_2 = k_1(x_1) = -x_1 - x_1^3$. This choice will determine the shape of the **rclf** for the entire second-order system, and it is clearly nonunique. We therefore emphasize the role of $k_1(x_1)$ as a design choice to be adjusted to achieve desired performance.

After we design $V_1(x_1)$ and $k_1(x_1)$ according to the above preliminary step, we are ready to construct an **rclf** V for the entire second-order system. We will look for a function $V_2(x_1, x_2)$ so that $V = V_1 + V_2$ is an **rclf** for our system. Our choice for V_2 will be based on the following calculation:

$$\begin{aligned} \max_w \dot{V} &\leq \max_w \dot{V}_1 + \max_w \dot{V}_2 \;\leq\; x_1 x_2 + x_1^4 + \max_w \dot{V}_2 \\ &\leq -x_1^2 + x_1\left[x_2 - k_1(x_1)\right] + \max_w \dot{V}_2 && (7) \end{aligned}$$

The second line follows from (6), and the third line shows the discrepancy between x_2 and our choice for $k_1(x_1)$. From this calculation we see that a simple choice for V_2 is $V_2 = \frac{1}{2}[x_2 - k_1(x_1)]^2$, and with this choice we obtain

$$\max_w \dot{V} \;\leq\; -x_1^2 + x_1\left[x_2 - k_1(x_1)\right] + \left[x_2 - k_1(x_1)\right]u + \max_w \Big[\left[x_2 - k_1(x_1)\right](w_2 - \dot{k}_1)\Big] \quad (8)$$

where $\dot{k}_1$ can be explicitly calculated as a function of x_1, x_2, and w_1. Clearly the right-hand side of (8) can be made negative by choice of u for all $(x_1, x_2) \neq (0,0)$, and it follows that V is an **rclf** for our system with $\Pi = \{(0,0)\}$. We thus conclude that the second-order system (4)–(5) is RGPS. One can show that if w_2 were not present, the **scp** would hold and we would therefore conclude that the system is RGAS.

We have found an **rclf** and thus determined that our system is stabilizable. The second major step in the design is to choose a stabilizing feedback law $u = k(x_1, x_2)$ according to the **rclf**. This step can be accomplished by manipulating the expression (8) using Young's inequality (or completing the squares) and then deciding on k by inspection; however, this method does not necessarily lead to a desirable feedback law (cf. Section 4). Alternatively one could solve the static nonlinear programming problem in [8] to obtain an optimal k.

Our above choice for the **rclf** was perhaps the simplest one, but it turns out that it can lead to feedback laws with unnecessarily large local gains in critical areas of the state space. Thus even if we choose an optimal feedback law according to the above **rclf**, the closed-loop behavior can be poor. One can try to improve performance by scaling V_1 and V_2, but the quadratic nature of V_2 as chosen above is an obstacle to any significant improvement. To construct a better V_2, we take advantage of our knowledge of the middle term in (7). In certain regions of the state space where this middle term is negative, we are free to choose $V_2 = 0$ and thus make $\dot{V}$ independent of u. Such a "flattened" **rclf** allows for greater flexibility in the choice for a feedback law, and as a consequence the required local gain can be greatly reduced. The proper choice for the **rclf** is therefore crucial in determining closed-loop performance.

The terminology "integrator backstepping" was introduced in [19], and the technique has evolved from the early results [4, 20, 21]. A generalization to the nonsmooth case is given in [22].

3.2 Unknown bounds

What if the set W above is compact but *unknown*? In [23] it was shown that the above tools can be applied to find an **rclf**, and in this case the bounded set Π will be *unknown*. We can therefore conclude that the system is RGS, but for a given feedback law we will have no *a priori* knowledge of the size of the resulting residual set. As an illustration, suppose $W = [-c, c]^2$ where c is an unknown positive constant. Then the calculation in (6) becomes

$$\max_w \dot{V}_1 \;=\; x_1 x_2 + c x_1^4 \tag{9}$$

Because c is unknown, we can no longer choose $k_1(x_1) = -x_1 - x_1^3$ as above; we must include a term which will dominate the c-term regardless of the value of c. It is clear from (9) that we need to choose $k_1(x_1) = -x_1 - \alpha(x_1)$ where $\alpha(x_1)$ is any smooth function which satisfies $\alpha(0) = 0$ and also $x_1\alpha(x_1)/x_1^4 \to \infty$ as $|x_1| \to \infty$. One possibility is $\alpha(x_1) = x_1^5$, and with this choice the calculation in (7) becomes

$$\begin{aligned}\max_w \dot{V} &\leq x_1 x_2 + c x_1^4 + \max_w \dot{V}_2 \\ &\leq -x_1^2 - x_1^6 + c x_1^4 + x_1\left[x_2 - k_1(x_1)\right] + \max_w \dot{V}_2 \end{aligned} \tag{10}$$

The combination of the first three terms in the right-hand side of (10) will be negative for x_1 sufficiently large, regardless of the value of c. Thus by choosing V_2 as above (either the quadratic one or the flattened one), we obtain an **rclf** for our system but with unknown Π.

What has been accomplished can be described in terms of input-to-state stability (ISS) with the disturbances w_1 and w_2 regarded as inputs. The relationship between the size of the residual set and c is determined by the ISS gain of the system, and this gain can be manipulated by adjusting the **rclf** and the feedback law. If we decrease the ISS gain (usually by increasing the control effort), we can reduce the effect of the disturbances on the system.

3.3 Adaptive backstepping

If we know more about the system uncertainties, we can make the stabilizing feedback laws less conservative and use smaller gains. A common situation is when all uncertainties occur as unknown *constant* parameters appearing linearly in the state equations; then we can modify the backstepping tool to include adaptation. In this way we can achieve asymptotic stabilization even without knowledge of bounds on the uncertainties. To illustrate, suppose w_1 and w_2 in the system (4)–(5) are constant unknown parameters. In this section we will design a feedback law $u = k(x_1, x_2, \widehat{w}_1, \widehat{w}_2)$ and parameter update laws for $\widehat{w}_1$ and $\widehat{w}_2$ using an adaptive backstepping procedure. The aim of this procedure is to *cancel* (rather than dominate) the uncertain terms appearing in $\dot{V}$.

We first pretend the state x_2 is the control and consider only the first state equation (4). If we choose $V_1 = \frac{1}{2}x_1^2 + \frac{1}{2}(\widehat{w}_1 - w_1)^2$ we obtain

$$\dot{V}_1 \;=\; x_1 x_2 + w_1 x_1^4 + (\widehat{w}_1 - w_1)\,\dot{\widehat{w}}_1 \tag{11}$$

We next find a control law $k_1(x_1, \widehat{w}_1)$ for the control x_2; a "certainty equivalence" choice would be $k_1 = -x_1 - \widehat{w}_1 x_1^3$ (exact cancellation would occur if $w_1 = \widehat{w}_1$). Substituting,

$$\dot{V}_1 \;=\; -x_1^2 + (\widehat{w}_1 - w_1)\,(\dot{\widehat{w}}_1 - x_1^4) + x_1\,[x_2 - k_1(x_1, \widehat{w}_1)] \tag{12}$$

We could choose an update law $\dot{\widehat{w}}_1 = x_1^4$ to eliminate the second term in (12), but this would lead to overparameterization. We instead call the would-be update law x_1^4 a *tuning function* and proceed with the design.

To keep the design simple, we will choose the quadratic form for V_2 (rather than the flattened form): $V_2 = \frac{1}{2}[x_2 - k_1(x_1, \widehat{w}_1)]^2 + \frac{1}{2}(\widehat{w}_2 - w_2)^2$. We compute $\dot{V}_2$ as follows:

$$\dot{V}_2 \;=\; [x_2 - k_1(x_1, \widehat{w}_1)]\,(u + w_2 - \dot{k}_1) + (\widehat{w}_2 - w_2)\,\dot{\widehat{w}}_2 \tag{13}$$

where $\dot{k}_1$ can be explicitly calculated as a function of x_1, x_2, w_1, $\widehat{w}_1$, and $\dot{\widehat{w}}_1$, with the unknown parameter w_1 appearing *linearly.* We now take $V = V_1 + V_2$ and add (13) to (12) to obtain

$$\dot{V} = -x_1^2 + [x_2 - k_1(x_1, \widehat{w}_1)]\,(u + x_1 + w_2 - \dot{k}_1) + (\widehat{w}_1 - w_1)\,(\dot{\widehat{w}}_1 - x_1^4) + (\widehat{w}_2 - w_2)\,\dot{\widehat{w}}_2 \tag{14}$$

At this point we would like to use part of the control u to cancel the matched terms $x_1 + w_2 - \dot{k}_1$. However, these terms contain the unknown parameters w_i and thus cannot be exactly cancelled. We instead "cancel" these terms using certainty equivalence, i.e., wherever we would like to use w_i we instead use $\widehat{w}_i$. Because the unknown parameters w_i appear linearly, the use of certainty equivalence produces errors which can be factored into the terms in (14) containing $\widehat{w}_1 - w_1$ and $\widehat{w}_2 - w_2$. Thus setting $u = v +$ certainty equivalence terms, we obtain

$$\dot{V} \;=\; -x_1^2 + [x_2 - k_1(x_1, \widehat{w}_1)]\,v + (\widehat{w}_1 - w_1)\,(\dot{\widehat{w}}_1 - \beta_1) + (\widehat{w}_2 - w_2)\,(\dot{\widehat{w}}_2 - \beta_2) \tag{15}$$

where the functions β_1 and β_2 contain no uncertainties. We can thus eliminate the last two terms in (15) by choosing update laws $\dot{\widehat{w}}_1 = \beta_1$ and $\dot{\widehat{w}}_2 = \beta_2$. We then choose v so that $\dot{V}$ is nonpositive and $\dot{V} < 0$ whenever $(x_1, x_2) \neq (0, 0)$ (for example, $v = -x_2 + k_1$). We conclude that the closed-loop fourth-order system is globally uniformly stable, and moreover that the states x_1 and x_2 converge to zero.

The adaptive backstepping tool evolved from the extended matching condition of [24]; further references will be discussed in Section 2.

3.4 Linear feedback

Suppose we change coordinates in the system (4)–(5) by setting $z_1 = x_1$ and $z_2 = x_2 + w_1 x_1^3$; then we obtain the following equivalent system:

$$\begin{aligned} \dot{z}_1 &= z_2 && (16)\\ \dot{z}_2 &= u + \phi(z_1, z_2, w_1, \dot{w}_1, w_2) && (17) \end{aligned}$$

where ϕ is a function resulting from the computation of $\dot{z}_2$. All of the uncertainties are now in ϕ and therefore satisfy the matching condition. However, the state z_2 is not available for feedback because it depends on the uncertainty w_1. Also, the derivative

$\dot{w}_1$ now appears in the state equations as an additional disturbance, and this derivative could be large even if w_1 itself is small. Nevertheless, the system written in the new form (16)–(17) is appealing because it allows us to use linear output feedback theory to obtain stabilizing linear feedback laws. These linear feedback laws achieve semi-global (rather than global) stability, that is, they guarantee a bounded region of attraction which can be made arbitrarily large by increasing the linear gain. One of the major obstacles to this approach is the peaking phenomenon caused by high gain [25, 26, 27]. It was shown in [28] that peaking can be tolerated in the observer using saturation in the controller, and this result became one of the main tools in subsequent designs [29, 30].

4 PROCEDURES

4.1 Strict feedback dominant systems

We can use the robust backstepping tool of Section 1 to find stabilizing controllers for systems of the form

$$\begin{aligned} \dot{x}_1 &= x_2 + \phi_1(x,u,t) && (18)\\ \dot{x}_2 &= x_3 + \phi_2(x,u,t) && (19)\\ &\vdots \\ \dot{x}_n &= u + \phi_n(x,u,t) && (20) \end{aligned}$$

Such a system is a chain of integrators perturbed by uncertain nonlinearities ϕ_i. The following assumption on the uncertainties ϕ_i allows us to apply the backstepping tool: for each ϕ_i we assume knowledge of a bounding function $\rho_i(x_1,\ldots,x_i)$ such that

$$|\phi_i(x,u,t)| \leq \rho_i(x_1,\ldots,x_i) \qquad (21)$$

for all x, u, and t. The restriction here is that the ith bounding function ρ_i depends only on the states x_1 through x_i; we call this the *strict feedback dominance* condition.

We first show that the above uncertainty structure can be formulated in terms of a disturbance w and a compact set W as required in the **rclf** definition. Indeed, we let $w = (w_1,\ldots,w_n)$ where $w_i = \phi_i(x,u,t)$ for each i. The set W will be of the form $W = W_1 \times \ldots \times W_n$ where $W_i = [-\rho_i, \rho_i]$. From (21) we see that each set W_i is a function of the states x_1 through x_i. The resulting set $W(x)$ is compact for each fixed x.

We construct an **rclf** for the system (18)–(20) by repeatedly applying the backstepping tool of Section 1. We first pretend x_2 is the control and design an **rclf** $V_1(x_1)$ and a feedback law $x_2 = k_1(x_1)$ according to the first state equation (18). We then construct a function $V_2(x_1, x_2)$ as in Section 1 such that $V_1 + V_2$ is an **rclf** for the second-order system (18)–(19) with x_3 now acting as the control variable. We then design a stabilizing feedback $x_3 = k_2(x_1, x_2)$ according to $V_1 + V_2$. We repeat this procedure with the addition of each integrator, at each step constructing a function V_i and a feedback law k_i, until we obtain an **rclf** $V = V_1 + \ldots + V_n$ for the entire system (18)–(20). Finally, we choose a stabilizing feedback law $u = k(x)$ according to the **rclf** V. The strict feedback dominance condition on the uncertainties guarantees that this recursive procedure is well-posed.

At each step in the recursive procedure, we have plenty of freedom in or choices for V_i and k_i, and these choices will be crucial in determining the performance of the system. For example, if we use the flattened form for each V_i instead of the quadratic form, we can obtain a feedback law with much lower local gains (especially when n is large). Such a feedback law can provide improved performance with less control effort as compared to a feedback law based on the quadratic V_i.

The flattened **rclf** was also instrumental in solving the problem of robustness with respect to unknown state measurement disturbances. By using the flattened form we can design a feedback law $k(x)$ such that the system with control $u = k(x+d)$ is (globally) ISS with respect to the input d (here d represents state measurement error due to inaccurate sensors or the use of an observer). This additional robustness is important when exact state measurements are unavailable.

The above design procedure first appeared in [31], and a recursive sliding mode version is given in [32]. A comprehensive version is described in [33], where it is shown that the strict feedback form is a nonlinear extension of the generalized matching condition of [34]. The design using a flattened **rclf** and its associated advantages are described in [35]. The modification of the design to achieve robustness with respect to state measurement error is given in [36].

4.2 Parametric strict feedback systems

Suppose each uncertainty ϕ_i in the system (18)–(20) can be written as

$$\phi_i(x,u,t) \quad = \quad \theta^T \gamma_i(x_1,\ldots,x_i) \tag{22}$$

where θ is a vector of *unknown constant parameters* and each γ_i is a *known* smooth function satisfying $\gamma_i(0)=0$. Note that similar to the bounding functions ρ_i of Section 1, the ith function γ_i depends only on the states x_1 through x_i. Systems with such uncertainties are said to be in *parametric strict feedback form.* For these systems we can repeatedly apply the adaptive backstepping tool of Section 3 to obtain a stabilizing controller, just as we repeatedly applied the robust backstepping tool in Section 1. At each step we call the would-be parameter update law a tuning function and propagate it through the rest of the design. The resulting controller consists of a feedback law plus one update law for each unknown parameter, so that the dynamic order of the controller is the same as the number of unknown parameters.

The original adaptive backstepping design procedure appeared in [37] along with geometric conditions for the existence of a coordinate transformation into parametric strict feedback form. The overparameterization in this original design was cut in half in [38] and completely eliminated by the introduction of tuning functions in [39].

In an alternative parameter estimation approach [40], we first apply the tools of Sections 1 and 2 to obtain a feedback law which guarantees boundedness in the absence of adaptation. We can then add parameter estimators to the system and use the parameter estimates in the control law to drive the state to the origin. Such estimation-based adaptive schemes have been developed in [41, 42].

5 CONCLUDING REMARKS

We have reviewed only several of many concepts, tools, and procedures currently employed in the design of robustly stabilizing feedback controls. The **rclf** appears to be a unifying concept for these designs. Robust and adaptive backstepping are effective tools for constructing both Lyapunov functions and stabilizing feedback laws for strict feedback systems. The flexibility of these tools creates the opportunity for performance improvement, such as with the flattened **rclf**. A natural further step is to address the issue of optimality, and we have mentioned some recent advances in that direction.

REFERENCES

[1] A. Isidori, *Nonlinear Control Systems.* Berlin: Springer Verlag, second ed., 1989.

[2] H. Nijmeijer and A. J. van der Schaft, *Nonlinear Dynamical Control Systems.* New York: Springer Verlag, 1990.

[3] Z. Artstein, "Stabilization with relaxed controls," *Nonlinear Anal.*, vol. 7, no. 11, pp. 1163–1173, 1983.

[4] J. Tsinias, "Sufficient Lyapunov-like conditions for stabilization," *Math. Control Signals Systems*, vol. 2, pp. 343–357, 1989.

[5] E. D. Sontag, "A 'universal' construction of Artstein's theorem on nonlinear stabilization," *Syst. Contr. Lett.*, vol. 13, no. 2, pp. 117–123, 1989.

[6] E. D. Sontag, "A Lyapunov-like characterization of asymptotic controllability," *SIAM J. Contr. Optimiz.*, vol. 21, no. 3, pp. 462–471, 1983.

[7] J. Tsinias, "Asymptotic feedback stabilization: A sufficient condition for the existence of control Lyapunov functions," *Syst. Contr. Lett.*, vol. 15, pp. 441–448, 1990.

[8] R. A. Freeman and P. V. Kokotovic, "Optimality of robust nonlinear feedback controls." In preparation, September 1993.

[9] M. J. Corless and G. Leitmann, "Continuous state feedback guaranteeing uniform ultimate boundedness for uncertain dynamic systems," *IEEE Trans. Automat. Contr.*, vol. 26, pp. 1139–1144, October 1981.

[10] B. R. Barmish, M. J. Corless, and G. Leitmann, "A new class of stabilizing controllers for uncertain dynamical systems," *SIAM J. Contr. Optimiz.*, vol. 21, pp. 246–255, 1983.

[11] Y. H. Chen and G. Leitmann, "Robustness of uncertain systems in the absence of matching assumptions," *Int. J. Contr.*, vol. 45, pp. 1527–1542, 1987.

[12] Y. H. Chen, "A new matching condition for robust control design," in *Proceedings of the American Control Conference*, (San Francisco, California), pp. 122–126, June 1993.

[13] S. T. Glad, "Robustness of nonlinear state feedback—a survey," *Automatica*, vol. 23, no. 4, pp. 425–435, 1987.

[14] J. Tsinias, "Optimal controllers and output feedback stabilization," *Syst. Contr. Lett.*, vol. 15, pp. 277–284, 1990.

[15] H. Hermes, "Asymptotically stabilizing feedback controls and the nonlinear regulator problem," *SIAM J. Contr. Optimiz.*, vol. 29, no. 1, pp. 185–196, 1991.

[16] A. Isidori and A. Astolfi, "Disturbance attenuation and H_∞-control via measurement feedback in nonlinear systems," *IEEE Trans. Automat. Contr.*, vol. 37, no. 9, pp. 1283–1293, 1992.

[17] T. Basar and P. Bernhard, *H_∞-Optimal Control and Related Minimax Design Problems.* Boston: Birkhäuser, 1991.

[18] R. A. Freeman and P. V. Kokotovic, "Design and comparison of globally stabilizing controllers for an uncertain nonlinear system," in *Systems, Models, and Feedback: Theory and Applications* (A. Isidori and T. J. Tarn, eds.), pp. 249–264, Boston: Birkhäuser, 1992.

[19] I. Kanellakopoulos, P. V. Kokotovic, and A. S. Morse, "A toolkit for nonlinear feedback design," *Syst. Contr. Lett.*, vol. 18, pp. 83–92, 1992.

[20] C. I. Byrnes and A. Isidori, "New results and examples in nonlinear feedback stabilization," *Syst. Contr. Lett.*, vol. 12, pp. 437–442, 1989.

[21] P. V. Kokotovic and H. J. Sussmann, "A positive real condition for global stabilization of nonlinear systems," *Syst. Contr. Lett.*, vol. 13, pp. 125–133, 1989.

[22] J.-M. Coron and L. Praly, "Adding an integrator for the stabilization problem," *Syst. Contr. Lett.*, vol. 17, pp. 89–104, 1991.

[23] I. Kanellakopoulos, "'Low-gain' robust control of uncertain nonlinear systems," Tech. Rep. UCLA/EE/CL 930224, University of California, Los Angeles, 1993.

[24] I. Kanellakopoulos, P. V. Kokotovic, and R. Marino, "An extended direct scheme for robust adaptive nonlinear control," *Automatica*, vol. 27, no. 2, pp. 247–255, 1991.

[25] H. J. Sussmann and P. V. Kokotovic, "The peaking phenomenon and the global stabilization of nonlinear systems," *IEEE Trans. Automat. Contr.*, vol. 36, no. 4, pp. 424–440, 1991.

[26] C. I. Byrnes and A. Isidori, "Asymptotic stabilization of minimum phase nonlinear systems," *IEEE Trans. Automat. Contr.*, vol. 36, no. 10, pp. 1122–1137, 1991.

[27] A. Saberi, P. V. Kokotovic, and H. J. Sussmann, "Global stabilization of partially linear composite systems," *SIAM J. Contr. Optimiz.*, vol. 28, no. 6, pp. 1491–1503, 1990.

[28] F. Esfandiari and H. K. Khalil, "Output feedback stabilization of fully linearizable systems," *Int. J. Contr.*, vol. 56, no. 5, pp. 1007–1037, 1992.

[29] A. Teel and L. Praly, "Tools for semi-global stabilization by partial state and output feedback." Submitted to *SIAM J. Contr. Optimiz.*, November 1992.

[30] Z. Lin and A. Saberi, "Robust semi-global stabilization of minimum-phase input-output linearizable systems via partial state and output feedback." Submitted to *IEEE Trans. Automat. Contr.*, May 1993.

[31] R. Marino and P. Tomei, "Robust stabilization of feedback linearizable time-varying uncertain nonlinear systems," *Automatica*, vol. 29, no. 1, pp. 181–189, 1993.

[32] J. J. E. Slotine and K. Hedrick, "Robust input-output feedback linearization," *Int. J. Contr.*, vol. 57, pp. 1133–1139, 1993.

[33] Z. Qu, "Robust control of nonlinear uncertain systems under generalized matching conditions," *Automatica*, vol. 29, no. 4, pp. 985–998, 1993.

[34] J. S. Thorp and B. R. Barmish, "On guaranteed stability of uncertain linear systems via linear control," *J. Optimiz. Theory App.*, vol. 35, pp. 559–579, 1981.

[35] R. A. Freeman and P. V. Kokotovic, "Design of 'softer' robust nonlinear control laws." To appear in *Automatica*, November 1993.

[36] R. A. Freeman and P. V. Kokotovic, "Global robustness of nonlinear systems to state measurement disturbances." To be presented at the 32nd IEEE Conference on Decision and Control, December 1993.

[37] I. Kanellakopoulos, P. V. Kokotovic, and A. S. Morse, "Systematic design of adaptive controllers for feedback linearizable systems," *IEEE Trans. Automat. Contr.*, vol. 36, no. 11, pp. 1241–1253, 1991.

[38] Z. P. Jiang and L. Praly, "Iterative designs of adaptive controllers for systems with nonlinear integrators," in *Proceedings of the 30th IEEE Conference on Decision and Control*, pp. 2482–2487, December 1991.

[39] M. Krstic, I. Kanellakopoulos, and P. V. Kokotovic, "Adaptive nonlinear control without overparameterization," *Syst. Contr. Lett.*, vol. 19, pp. 177–185, 1992.

[40] L. Praly, G. Bastin, J.-B. Pomet, and Z. P. Jiang, "Adaptive stabilization of nonlinear systems," in *Foundations of Adaptive Control* (P. V. Kokotovic, ed.), pp. 347–434, Berlin: Springer Verlag, 1991.

[41] M. Krstic and P. V. Kokotovic, "Adaptive nonlinear control with nonlinear swapping." To be presented at the 32nd IEEE Conference on Decision and Control, December 1993.

[42] M. Krstic and P. V. Kokotovic, "Observer-based schemes for adaptive nonlinear state-feedback control." To appear in *Int. J. Contr.*, 1993.

Disturbance attenuation with steady-state constraints for nonlinear systems *

A. Isidori*° *and* T.J.Tarn°

°Department of Systems Science and Mathematics,
Washington University, St.Louis, MO 63130
and
*Dipartimento di Informatica e Sistemistica,
Università di Roma "La Sapienza", 00184 Rome

I. PROBLEM STATEMENT

Consider a nonlinear system described by equations of the form

$$\dot{x} = f(x) + g_1(x)w + g_2(x)u. \tag{1}$$

$$z = \begin{pmatrix} z_1 \\ z_2 \\ u \end{pmatrix} = \begin{pmatrix} h_1(x) \\ h_2(x) \\ u \end{pmatrix}, \tag{2}$$

As usual, $u \in \mathbb{R}^m$ denotes a *control input*, while $w \in \mathbb{R}^r$ denotes an *exogenous input*, which includes both *disturbances* (to be rejected) and *commands* (to be tracked). The state x of this system is defined on a neighborhood of the origin in $\mathbb{R}^n$. The second one of these equations defines a *controlled output* and it is assumed that both x and z_1 are available for feedback.

The mappings $f(x), g_1(x), g_2(x), h_1(x), h_2(x)$ are smooth mappings (i.e. mappings of class C^k for some sufficiently large k) defined in a neighborhood of the origin in $\mathbb{R}^n$. It is also assumed that $f(0) = 0, h_1(0) = 0, h_2(0) = 0$.

In this paper, we study the problem of disturbance attenuation (with local internal stability) with constraints on the steady state response to a sinusoidal input of fixed frequency ω_0. The problem in question is to find, if possible, an integer ν and a feedback law of the form

$$\begin{aligned} \dot{\xi} &= \eta(\xi, x, z_1) \\ u &= \theta(\xi, x, z_1) \end{aligned}$$

where $\xi \in \mathbb{R}^\nu$, such that:

(a) the closed loop system has a locally exponentially stable equilibrium at $(x, \xi) = (0, 0)$,

(b) the L_2-gain of the closed loop system from the input w to the output z is less than or equal to a prescribed number γ,

---

*Work supported in part by NSF under grant ECS-9208306 and by MURST.

(c) the steady state response of the output z_1 of the closed loop system to any sufficiently small sinusoidal exogenous input of fixed frequency ω_0 (which is well-defined because of property (a)) is zero.

Note also that, in view of an interesting analogy established in [11], the requirement (b) could be replaced by the following (perhaps more appealing) one:

(b') the steady state response of the output z of the closed loop system to any sufficiently small periodic input w of arbitrary period T (which is well-defined because of property (a)) has an *RMS* value which does exceed γ times the *RMS* value of the corresponding input (see eq. (29) of [11]).

For a linear system, requirements (b') and (c) are precisely the requirements of having a closed loop system whose transfer function matrix $T_{zw}(s)$ from the exogenous input w to the controlled output z satisfies

$$\|T_{zw}(j\omega)\| \leq \gamma, \quad \text{for all} \quad \omega \in I\!R$$

with constraints of the form

$$(\, I \quad 0\,)\, T_{zw}(j\omega_0) = 0.$$

For an interpretation of this design problem as a *robust performance problem*, the reader is referred to [1] and [2].

II. THE CONSTRAINT ON THE STEADY-STATE RESPONSE

The requirements indicated in (a) and (b), namely to obtain a locally exponentially stable closed loop system yielding zero steady-state response to a sinusoidal exogenous input, indeed characterize a problem of *output regulation*, for a plant modeled by equations of the form (1), in which the *tracking error* is defined by the map

$$e = h_1(x)$$

and in which the exogenous input w is generated by an *exosystem* of the form

$$\dot{w} = Sw \tag{3}$$

and S is a skew-symmetric matrix.

A design problem of this type has been thoroughly investigated in the papers [12] and [9]. The possibility (and the difficulties connected to the problem) of achieving *robust regulation* (namely, to obtain a zero steady state error also in the presence of plant parameters uncertainties) was discussed in the paper [6] and, more recently, in the papers [7] and [5]. In particular, the earlier paper [6] deals with the problem of robust regulation in the case of *constant* exogenous inputs, that is in the case where

$$\dot{w} = 0,$$

and shows that any *integral control*, that is any feedback law of the form

$$\begin{aligned} \dot{\xi} &= e \\ u &= \theta(x, \xi), \end{aligned}$$

so long as the control $u = \theta(x, \xi)$ locally exponentially stabilizes the closed loop system, induces robust output regulation. Unfortunately, the simple arguments showing that a controller including an *internal model* of the exosystem is a robust controller no longer hold, for a nonlinear plant, if the exosystem is nontrivial, i.e. if the exogenous inputs are non-constant. A more elaborate analysis is actually needed and some severe restrictions apply, as shown in [7] and [5].

In this section, we examine the problem of when a controller of the form

$$\begin{aligned} \dot{\xi} &= F\xi + Ge \\ u &= \theta(x, \xi), \end{aligned} \tag{4}$$

(where $F \in I\!R^{\nu \times \nu}$ is a matrix playing the role of an internal model of the exogenous inputs) induces robust regulation. The results we obtain are indeed very similar to some results of [7] and [5], though the derivation we provide is somewhat different and, in some sense, more straightforward.

To this purpose, we need to introduce some auxiliary notation. Let k be a fixed integer and let $\mathcal{P}_k$ denote the set of all polynomials in r of degree k in the indeterminates $w_1, w_2, \dots, w_r$ with coefficients in $I\!R$. $\mathcal{P}_k$ is indeed a finite-dimensional vector space over $I\!R$. Consider the mapping

$$\begin{aligned} D_{Sw}^{(k)} : \mathcal{P}_k &\rightarrow \mathcal{P}_k \\ \mathbf{p} &\mapsto \frac{\partial \mathbf{p}}{\partial w} Sw. \end{aligned}$$

Clearly $D_{Sw}^{(k)}$ is a $I\!R$-linear mapping. Let μ denote the degree of the minimal polynomial of $D_{Sw}^{(k)}$ and let $M^{(k)} \in I\!R^{\mu \times \mu}$ be a matrix having the same minimal polynomial as $D_{Sw}^{(k)}$. Then, it is not difficult to check that the following property holds.

Lemma II.1 *Let $g \in I\!R^{\mu}$ be nonzero. Suppose for each μ-tuple $(\eta_1, \dots, \eta_\mu) \in \mathcal{P}_k \times \cdots \times \mathcal{P}_k$ there exists a μ-tuple $(\sigma_1, \dots, \sigma_\mu) \in \mathcal{P}_k \times \cdots \times \mathcal{P}_k$ and $\gamma \in \mathcal{P}_k$ satisfying*

$$\begin{pmatrix} \eta_1 \\ \cdots \\ \eta_\mu \end{pmatrix} = D_{Sw}^{(k)} \begin{pmatrix} \sigma_1 \\ \cdots \\ \sigma_\mu \end{pmatrix} - M^{(k)} \begin{pmatrix} \sigma_1 \\ \cdots \\ \sigma_\mu \end{pmatrix} + g\gamma.$$

Then, $(\eta_1, \dots, \eta_\mu) = (0, \dots, 0)$ implies $\gamma = 0$.

Moreover, recall the following standard result of linear multivariable control theory.

Lemma II.2 *Suppose $(M^{(k)}, g)$, with $M^{(k)} \in I\!R^{\mu \times \mu}$ and $g \in I\!R^{\mu \times 1}$, is controllable. Suppose (A, B), with $A \in I\!R^{n \times n}$ and $B \in I\!R^{n \times m}$ is stabilizable. Let $C \in I\!R^{n \times p}$ and suppose*

$$\mathit{rank} \begin{pmatrix} A - \lambda I & B \\ C & 0 \end{pmatrix} = n + p$$

for each eigenvalue λ of $M^{(k)}$. Set $\nu = \mu p$ and let $S^{(k)} \in I\!R^{\nu \times \nu}$ and $G \in I\!R^{\nu \times p}$ be defined as follows

$$S^{(k)} = \mathit{diag}\{M^{(k)}, \dots, M^{(k)}\}, \quad G = \mathit{diag}\{g, \dots, g\}. \tag{5}$$

Then the pair

$$\begin{pmatrix} A & 0 \\ GC & S^{(k)} \end{pmatrix}, \begin{pmatrix} B \\ 0 \end{pmatrix}$$

is stabilizable.

This result can be used, as one may easily understand, in the following way. Set

$$A = \left(\frac{\partial f}{\partial x}\right)_{x=0}, \quad C_1 = \left(\frac{\partial h_1}{\partial x}\right)_{x=0}, \quad B_2 = g_2(0).$$

Suppose (A, B_2) is stabilizable and

$$rank \begin{pmatrix} A - \lambda I & B_2 \\ C_1 & 0 \end{pmatrix} = n + p$$

for each eigenvalue λ of $M^{(k)}$ (where p denotes the number of rows of C_1). Then the composite system

$$\begin{aligned} \dot{x} &= f(x) + g_1(x)w + g_2(x)u \\ \dot{\xi} &= S^{(k)}\xi + Gh_1(x) \end{aligned} \tag{6}$$

is locally exponentially stabilizable, about the equilibrium $x = 0$, by means of a feedback law of the form

$$u = \theta(x, \xi). \tag{7}$$

If the closed loop system (6)-(7) is locally exponentially stable, each (sufficiently small) sinusoidal exogenous input produces a steady state response. Following [12], observe that system (6)-(7) driven by the exosystem (3), i.e. the system

$$\begin{aligned} \dot{x} &= f(x) + g_1(x)w + g_2(x)\theta(x, \xi) \\ \dot{\xi} &= S^{(k)}\xi + Gh_1(x) \\ \dot{w} &= Sw \end{aligned} \tag{8}$$

has a center manifold, at $(x, \xi, w) = (0, 0, 0)$ which can be expressed as a graph of a mapping

$$w \mapsto \begin{pmatrix} \pi(w) \\ \sigma(w) \end{pmatrix}.$$

By definition, the latter satisfies

$$\begin{aligned} \frac{\partial \pi}{\partial w} Sw &= f(\pi(w)) + g_1(\pi(w))w + g_2(\pi(w))\theta(\pi(w), \sigma(w)) \\ \frac{\partial \sigma(w)}{\partial w} Sw &= S^{(k)}\sigma(w) + Gh_1(\pi(w)). \end{aligned} \tag{9}$$

As shown in [12], any exogenous input $w(t)$ generated by the exosystem (3) produces a steady-state response of the form

$$e(t) = h_1(\pi(w(t))).$$

Using Lemma II.1 and a well-know result on polynomial approximation of center manifolds, one can easily prove the following (see also [7]).

Proposition II.1 *The mapping $\pi(w)$ is such that*

$$h_1(\pi(w)) = 0 \mod \mathcal{P}_k.$$

Thus, in particular, if $h_1(\pi(w))$ is a polynomial of degree less than or equal to k, the steady state response of (6)-(7), to any exogenous input $w(t)$ generated by the exosystem (3), is identically zero.

III. ACHIEVING DISTURBANCE ATTENUATION

In the previous section, we have shown that a feedback law of the form

$$\begin{aligned} \dot{\xi} &= S^{(k)}\xi + Gz_1 \\ u &= \theta(x,\xi), \end{aligned} \tag{10}$$

with $S^{(k)}$ and G matrices defined as in (5), if the corresponding closed loop system is locally exponentially stable (i.e. if requirement (a) holds) and an appropriate additional condition is met (namely, if a solution $\pi(w)$ of (9) is such that $h_1(\pi(w))$ is a polynomial of degree less than or equal to k), is able to fulfill the requirement (c) of the design problem under consideration.

Thus, we proceed to discuss the problem of when there exists a feedback law of the form (10) able to fulfill the two requirements (a) and (b). To this end, note that the interconnection of (1) and the first equation in (10) is a particular case of a composite system modeled by equations of the form

$$\begin{aligned} \begin{pmatrix} \dot{x} \\ \dot{\xi} \end{pmatrix} &= \begin{pmatrix} f(x) \\ p(x) + S\xi \end{pmatrix} + \begin{pmatrix} g_1(x) \\ q_1(x) \end{pmatrix} w + \begin{pmatrix} g_2(x) \\ q_2(x) \end{pmatrix} u \\ z &= \begin{pmatrix} h(x) \\ u \end{pmatrix} \end{aligned} \tag{11}$$

(for $p(x) = Gh_1(x)$, $S = S^{(k)}$, $q_1(x) = 0$, $q_2(x) = 0$ and $h^{\mathrm{T}}(x) = (\, h_1^{\mathrm{T}}(x) \quad h_2^{\mathrm{T}}(x)\,)^{\mathrm{T}}$).

The problem of disturbance attenuation, with local exponential internal stability, via state feedback for systems modeled by equations of this type was discussed in the paper [13]. In particular, one of the main results of this paper was to show how existence of a solution of the problem in question can be related to the existence of a solution of the Hamilton-Jacobi-Isaacs inequality

$$V_x f(x) + h^{\mathrm{T}}(x)h(x) + \frac{1}{4}V_x\left(\frac{1}{\gamma^2}g_1(x)g_1^{\mathrm{T}}(x) - g_2(x)g_2^{\mathrm{T}}(x)\right)V_x^{\mathrm{T}} \le 0, \tag{12}$$

arising in the problem of disturbance attenuation for the plant (1) with controlled output z satisfying

$$z^{\mathrm{T}}z = h^{\mathrm{T}}(x)h(x) + u^{\mathrm{T}}u. \tag{13}$$

We summarize hereafter some of the results established in [13]. Let $V(x)$ be a smooth solution of (12) and set

$$\alpha_1(x) = \frac{1}{2\gamma^2}g_1^{\mathrm{T}}(x)V_x^{\mathrm{T}}, \qquad \alpha_2(x) = -\frac{1}{2}g_2^{\mathrm{T}}(x)V_x^{\mathrm{T}}. \tag{14}$$

Set also

$$F(x,\xi) = \begin{pmatrix} f(x) \\ p(x) + S\xi \end{pmatrix}, \quad G_1(x) = \begin{pmatrix} g_1(x) \\ q_1(x) \end{pmatrix}, \quad G_2(x) = \begin{pmatrix} g_2(x) \\ q_2(x) \end{pmatrix}.$$

Let $\xi = \lambda(x)$ be a smooth function, vanishing at $x = 0$, and suppose the vector field

$$F^*(x,\xi) = F(x,\xi) + G_1(x)\alpha_1(x) + G_2(x)\alpha_2(x) \tag{15}$$

is *tangent* to the manifold

$$M = \{(x, \xi) : \xi = \lambda(x)\}$$

that is, suppose the function $\lambda(x)$ is a solution of the partial differential equation

$$\left(\frac{\partial \lambda(x)}{\partial x} \quad -I \right) F^{*}(x, \xi)_{\xi = \lambda(x)} = 0. \tag{16}$$

Using the function $\lambda(x)$ thus characterized, define

$$\delta_1(x) = \left(\frac{\partial \lambda(x)}{\partial x} \quad -I \right) G_1(x), \quad \delta_2(x) = \left(\frac{\partial \lambda(x)}{\partial x} \quad -I \right) G_2(x). \tag{17}$$

Finally, set

$$A = \left(\frac{\partial f}{\partial x} \right)_{x=0}, \quad P = \left(\frac{\partial p}{\partial x} \right)_{x=0}, \quad B_2 = g_2(0), \quad D_2 = q_2(0).$$

Then we have the following.

Proposition III.1 *Suppose S is a skew-symmetric matrix. Suppose the pair*

$$\begin{pmatrix} A & 0 \\ P & S \end{pmatrix}, \quad \begin{pmatrix} B_2 \\ 0 \end{pmatrix}$$

is stabilizable. Suppose the matrix A has no eigenvalues on the imaginary axis.

Assume the following:

(i) *there exists a smooth solution $V(x) > 0$ of the Hamilton-Jacobi-Isaacs inequality (12),*

(ii) *there exists a smooth solution $\lambda(x)$ of the partial differential equation (16),*

(iii) *the functions $\delta_1(x)$ and $\delta_2(x)$ defined by (17) are such that*

$$\gamma^2 \delta_2(x) \delta_2^{\mathrm{T}}(x) \geq \delta_1(x) \delta_1^{\mathrm{T}}(x) \tag{18}$$

for all x.

Then the feedback law

$$u = \alpha_2(x) + \delta_2^{\mathrm{T}}(x)(\xi - \lambda(x)) \tag{19}$$

solves the problem of disturbance attenuation, with local exponential internal stability, for system (11).

IV. CONCLUSIONS

The results developed in the previous two sections provide the necessary means to solve the problem indicated at the beginning of the paper. In fact, observe that, since S is skew-symmetric, one can choose $M^{(k)}$ to be skew-symmetric as well. Thus also $S^{(k)}$ is skew-symmetric.

Suppose the pair (A, B_2) is stabilizable. Suppose the matrix

$$\begin{pmatrix} A - \lambda I & B_2 \\ C_1 & 0 \end{pmatrix}$$

has linearly independent rows for each eigenvalue λ of $M^{(k)}$. Suppose the matrix A has no eigenvalues on the imaginary axis. Finally, suppose there exists functions $V(x)$ and $\lambda(x)$ satisfying the conditions (i)-(ii)-(iii) of Proposition III.1.

Then, the controller

$$\begin{aligned} \dot{\xi} &= S^{(k)}\xi + Gz_1 \\ u &= \alpha_2(x) + \delta_2^{\mathrm{T}}(x)(\xi - \lambda(x)) \end{aligned}$$

is such that requirements (a) and (b) hold. Moreover, any (sufficiently small) exogenous input $w(t)$ produces a response $z_1(t)$ which asymptotically converges towards a steady state response $z_1^{ss}(t)$ which can be expressed as

$$z_1^{ss}(t) = R(w(t))$$

where the function

$$\begin{array}{rcl} R : I\!R^r & \rightarrow & I\!R^p \\ w & \mapsto & h_1(\pi(w)) \end{array}$$

satisfies

$$\lim_{w \to 0} \frac{\|R(w)\|}{\|w\|^k} = 0.$$

If, in particular, there exists a solution $(\pi(w), \sigma(w))$ of (9) such that $h_1(\pi(w))$ is a polynomial of degree less than or equal to k, then also requirement (c) holds.

V. ACKNOWLEDGMENTS

The authors would like to thank P.Khargonekar for having called their attention on the interest of the problem of constrained disturbance attenuation for nonlinear systems and having provided them with the preprint of [1]. They also indebted to thank C.Scherer for many interesting discussions that substantially contributed to the developments discussed in the second part of the paper.

REFERENCES

[1] J. Abedor, K. Nagpal, P.P. Khargonekar, K. Poolla, Robust regulation with an H_∞ constraint, In *Control of Uncertain Systems*: 95–110, CRC Press, 1991.

[2] J. Abedor, K. Nagpal, P.P. Khargonekar, K. Poolla, Robust regulation in the presence of norm-bounded uncertainty, preprint, 1993.

[3] J. Ball, J.W. Helton, and M.L. Walker, H_∞ control for nonlinear systems via output feedback, *IEEE Trans. Autom. Control*, **AC-38**: 546–559, 1993

[4] T. Basar and P. Bernhard, *H_∞-optimal control and related Minimax design problems*, Birkhauser, 1990.

[5] F.Delli Priscoli, Robust tracking for polynomial plants, *Proc. of 2nd European Control Conf.*: Groeningen, The Netherlands, June 1993.

[6] B.A.Francis, W.M.Wonham, The internal model principle of control theory, *Automatica*, **12**: 457–465, 1977.

[7] J.Huang, C.F.Lin, On a robust nonlinear servomechanism problem, In *Proc. of 30th IEEE Conf. Decision and Control*: 2529–2530, Brighton, England, December 1991.

[8] J.Huang, W.J.Rugh, On the nonlinear multivariable servomechanism problem, *Automatica*, **26**: 963–972, 1990.

[9] J.Huang, W.J.Rugh, An approximation method for the nonlinear servomechanism problem, *IEEE Trans. Autom. Control*, **AC-37**: 1395–1398, 1992.

[10] A. Isidori, Feedback control of nonlinear systems, In *Proc. of 1st European Control Conf.*: 1000–1012, Grenoble, France, July 1991.

[11] A. Isidori and A. Astolfi, Disturbance attenuation and H_∞ control via measurement feedback in nonlinear systems, *IEEE Trans. Autom. Control*, **AC-37**:1283–1293, 1992.

[12] A.Isidori, C.I.Byrnes, Output regulation of nonlinear systems, *IEEE Trans. Autom. Control*, **AC-35**: 131–140, 1990.

[13] A.Isidori, T.J.Tarn, Dissipation inequalities for a class of composite nonlinear systems, with application to constrained disturbance attenuation, In *Proc. of 32th IEEE Conf. Decision and Control*: San Antonio, TX, December 1993.

[14] C. Scherer, H_∞-control by state feedback for plants with zeros on the imaginary axis, *SIAM J. Control and Opitimiz.*, **30**: 123–142, 1992.

[15] A.A. Stoorvogel, *The H_∞ control problem: a state space approach*, Prentice-Hall Englewood-Cliffs, 1992.

[16] T.Sugie, S.Hara, H_∞ suboptimal control with boundary constraints, *Syst. and Contr. Lett.*, **13**:93–99, 1989.

[17] A.J. Van der Schaft, A state-space approach to nonlinear H_∞ control, *Syst. and Contr. Lett.*, **16**:1–8, 1991.

[18] A.J. Van der Schaft, L_2-gain analysis of nonlinear systems and nonlinear H_∞ control, *IEEE Trans. Autom. Control*, **AC-37**:770–784, 1992.

Discrete Time Sliding Mode Control in Nonlinear Sampled-Data Systems

Wu-Chung Su, Ümit Özgüner, Sergey V. Drakunov

Department of Electrical Engineering
The Ohio State University
2015 Neil Avenue
Columbus, OH 43210

Abstract

This paper deals with sliding mode control problems for nonlinear sampled-data systems. The nonlinearities and uncertainties can be approximated by the past state values if they are bounded and smooth. With the approximate information, the proposed sliding mode control law can steer the state into a boundary layer of the prescribed sliding surface with thickness to the order of $O(T^2)$, where T is the sampling period. Furthermore, chattering in the control will be smoothed as long as the $O(T^2)$ accuracy is maintained.

I. Introduction

Consider a nonlinear system of the form

$$\dot{x} = f(x) + g(x)u \tag{1}$$

with a prescribed sliding manifold

$$\mathcal{S} = \{x|s(x) = 0\}. \tag{2}$$

The input u, the state x, and the vector s are m, n, and m dimensional respectively; f and g are differentiable functions with appropriate dimensions. The intent of "sliding mode" control is to steer the state onto and maintain it on the sliding manifold, in which the system will possess desired properties such as stability and robustness. The continuous time, variable structure approach leads to a discontinuous control law in each component of u that switches between two functions u_i^+ and u_i^-, $i = 1, \cdots, m$, according to the sign of $s(x)$. When the sliding mode occurs, the resultant system can be viewed as lower dimensional and the dynamic equation of the sliding surface is

$$\dot{s}(x) = G(x)f(x) + G(x)g(x)u. \tag{3}$$

It is well known [1] that the "equivalent control" u^{eq} (obtained by setting $\dot{s} = 0$) is a key variable in the controller design. Here,

$$u^{eq}(t) = -[G(x)g(x)]^{-1}G(x)f(x), \tag{4}$$

where $G(x) = \frac{\partial}{\partial x}s(x)$ is the $m \times n$ Jacobian matrix of $s(x)$. According to [1], for sliding mode to exist in $\mathcal{S}$, it is necessary that the equivalent control exists and that each component $u_i^{eq}(t)$ is bounded between the values u_i^+ and u_i^-. With the assumption of infinite switching frequency, the discontinuous control system (1) in sliding mode behaves as a continuous one.

In a real sampled-data system, switching will occur at a finite frequency. Therefore, a direct translation of continuous time variable structure control design for discrete implementation will lead to severe chattering problems. To alleviate chattering, a discrete time approach, taking the sample/hold effects into account is necessary. We introduce the following definition to analyze this effect in a sampled-data system with sampling period T.

Definition 1 *The magnitude of a vector $w \in R^r$ is said to be of order $O(T^q)$ if there exists a constant M, $0 \leq M < \infty$ such that each component of w satisfies*

$$w_i \leq MT^q, \quad i = 1, \cdots, r; \quad q \in R.$$

Note that the constant M is independent of the vector w and the sampling period T.

In this paper, we consider the dynamics (1) in discrete time. Applying a zero-order-hold to the control yields the system representation

$$x_{k+1} = x_k + \int_{kT}^{(k+1)T} f(x)dt + \int_{kT}^{(k+1)T} g(x)dt\, u_k. \tag{5}$$

Furthermore, the equation (3), showing the dynamics of the sliding surface can be written as,

$$s_{k+1} = s_k + \int_{kT}^{(k+1)T} G(x)f(x)dt + \int_{kT}^{(k+1)T} G(x)g(x)dtu_k. \tag{6}$$

The discrete time equivalent control is obtained by setting $s_{k+1} = 0$ [2], which leads to

$$u_k^{eq} = -[\int_{kT}^{(k+1)T} G(x)g(x)dt]^{-1}[s_k + \int_{kT}^{(k+1)T} G(x)f(x)dt]. \tag{7}$$

Let the admissible control be bounded between some upper and lower limit values. Denote the allowable range by $\mathcal{U}$. For (4) and (7) to be valid for controller design, we apply the following assumptions:

Assumption 1

1. *The $m \times m$ matrix $G(x)g(x)$ is nonsingular for all x.*
2. *The $m \times m$ matrix $\int_{kT}^{(k+1)T} G(x)g(x)dt$ is nonsingular for all k.*
3. *The continuous time equivalent control $u^{eq}(t) \in \mathcal{U}$.*

It can be observed that u_k^{eq} will approach $u^{eq}(t)$ as $T \to 0$ when $s_k = 0$. Since the continuous time equivalent control $u^{eq}(t) \in \mathcal{U}$, the discrete time equivalent control u_k^{eq} will be admissible when the state is in the vicinity of $\mathcal{S}$. Note in (7) that $u_k^{eq} = O(\frac{1}{T})$ if x_k is far from $\mathcal{S}$ and it is usually nonadmissible.

Since the control in the sampled data system will show jumps from the continuous time point of view, the formal concept of continuity is not compatible with the notion of discrete time control. In order to make the analogy properly, let us introduce the following definition:

Definition 2 *The discrete time control law u_k is said to be equivalent to discontinuous if $\Delta u_k = O(1)$, continuous if $\Delta u_k = O(T)$, smooth if $\Delta^2 u_k = O(T^2)$, where Δ denotes the backward differencing operator $1 - z^{-1}$.*

To avoid a cumbersome description, the "continuity" of discrete time control to appear in the following text will refer to its equivalent meaning as defined above.

A considerable amount of research in discrete time sliding mode has been conducted [3, 4, 5, 6, 7]. The major difficulty is the necessity of having information about the system for calculating the discrete time equivalent control u_k^{eq} which drives the state to the sliding manifold in one step. If neglected, the external disturbances and possible parameter variations will force the state out of the sliding manifold leading to a boundary layer of order $O(T)$.

In [7], we have developed a method applied to linear systems to alleviate the above problem. Here, we extend it to nonlinear systems. We assume the nonlinearities and uncertainties are bounded and smooth, in this case the past values of the states can be utilized to predict the effect. This approach allows us to keep the state in the $O(T^2)$ vicinity of the sliding manifold implying removal of the chattering phenomena. The switching nature of the control only occurs outside the boundary layer. The proposed method is well suited to nonlinear sampled-data systems with a prescribed sliding manifold. Considerable robustness against both external and internal uncertainties is obtained.

II. Sliding Mode in a Sampled-Data System

We first consider the nonlinear sampled-data system (5) without disturbances. Introduce the following notation:

$$\begin{aligned} F_k &= \int_{kT}^{(k+1)T} G(x)f(x)dt \\ B_k &= \int_{kT}^{(k+1)T} G(x)g(x)dt \end{aligned} \tag{8}$$

so that (6) can be written as

$$s_{k+1} = s_k + F_k + B_k\, u_k. \tag{9}$$

F_k and B_k usually cannot be evaluated exactly since they depend on the future value of the state. Nevertheless, with some mild smoothness condition on the functions $G(x)$, $f(x)$, and $g(x)$, they can be approximated by the following expressions:

$$\begin{aligned} \bar{F}_k &= G(x_k)f(x_k)T \\ \bar{B}_k &= G(x_k)g(x_k)T. \end{aligned} \tag{10}$$

The legitimacy of the approximation (10) bases on the assumptions:

Assumption 2

1. *The functions $f(x)$, $g(x)$, and $G(x)$ are bounded and smooth; $\frac{\partial}{\partial x}[G(x)f(x)]$ and $\frac{\partial}{\partial x}[G(x)g(x)]$ are bounded.*
2. *There exist positive finite numbers m_g, M_g, M_f, and a nonnegative number m_f such that*

$$\begin{aligned} 0 < m_g \leq \|G(x)g(x)\| \leq M_g \\ 0 \leq m_f \leq \|G(x)f(x)\| \leq M_f \ \textit{ for all } x. \end{aligned}$$

Introduce the notation:

$$\begin{aligned} e_{\bar{F}_k} &= F_k - \bar{F}_k, \quad \bar{e}_{\bar{F}_k} = \bar{F}_k - \bar{F}_{k-1} \\ e_{\bar{B}_k} &= B_k - \bar{B}_k, \quad \bar{e}_{\bar{B}_k} = \bar{B}_k - \bar{B}_{k-1}. \end{aligned} \tag{11}$$

With Assumption 2, it can be proved (see Appendix) that there exist finite positive numbers M_F and M_B satisfying

$$\begin{aligned} \|e_{\bar{F}_k}\| \leq M_F T^2, \quad \|\bar{e}_{\bar{F}_k}\| \leq \tfrac{M_F}{2}T^2 \\ \|e_{\bar{B}_k}\| \leq M_B T^2, \quad \|\bar{e}_{\bar{B}_k}\| \leq \tfrac{M_B}{2}T^2. \end{aligned} \tag{12}$$

We obtain an approximation for the equivalent control (7)

$$u_k^{in} = -\bar{B}_k^{-1}(s_k + \bar{F}_k), \tag{13}$$

where $\bar{B}_k$ is invertible by Assumption 1.

Since the magnitude of $\bar{B}_k$ and $\bar{F}_k$ are both $O(T)$, s_k has to be at least as small as $O(T)$ for u_k^{in} to be admissible. In other words, the state should lie within a specific boundary layer of the sliding surface so that $u_k^{in} \in \mathcal{U}$. In order to define the boundary layer of the sliding surface, let us adopt the following notation:

$\mathcal{S}_T = \{x_k | u_k^{in} \in \mathcal{U}\}$ is the $O(T)$ boundary layer of $\mathcal{S}$.

We propose the variable structure control law

$$u_k = \begin{cases} -\bar{B}_k^{-1}(s_k + \bar{F}_k) = u_k^{in} & \text{when } x_k \in \mathcal{S}_T \\ -\bar{B}_k^{-1}(K sgn(s_k) + \bar{F}_k) = u_k^{out} & \text{otherwise.} \end{cases} \tag{14}$$

Note that u_k^{out} can always be made admissible by an appropriate choice of a positive definite matrix K since the second term $-\bar{B}_k^{-1}\bar{F}_k$ is actually the continuous time equivalent control evaluated at $t = kT$, which is admissible by assumption. If the state is out of the boundary layer $\mathcal{S}_T$, we apply the same control as continuous time design u_k^{out}. The sampling period T and the matrix K have to be chosen such that the next state approaches $\mathcal{S}_T$ without overshoot. This can be referred to as the "reaching phase". Chattering can occur in the reaching phase if there is partial sliding in some elements of the sliding surface vector s_k.

Once the state enters $\mathcal{S}_T$, the control u_k^{in} will further steer it into an $O(T^2)$ boundary layer and retain it there. Robustness against both external and internal uncertainties can then be acquired with sufficient accuracy. We will show that the proposed control in (14) will be continuous when $s_k = O(T^2)$.

Lemma 1 *The control u_k in (14) will lead to $s_{k+1} = O(T^2)$ if $x_k \in \mathcal{S}_T$.*

Proof. Since the state is in $\mathcal{S}_T$, the control u_k^{in} is employed. Direct substitution of u_k^{in} into (9) yields

$$s_{k+1} = e_{\bar{F}_k} - e_{\bar{B}_k}\bar{B}_k^{-1}(s_k + \bar{F}_k).$$

With $\bar{B}_k = O(T)$, $s_k = O(T)$, $\bar{F}_k = O(T)$, $e_{\bar{F}_k} = O(T^2)$, and $e_{\bar{B}_k} = O(T^2)$, it follows $s_{k+1} = O(T^2)$. □

Lemma 2 *The control u_k in (14) is continuous if $s_{k-1} = O(T^2)$.*

Proof. According to Lemma 1, $s_{k-1} = O(T^2)$ implies $u_{k-1} = u_{k-1}^{in}$ and $s_k = O(T^2)$, which leads to $s_k = O(T^2)$ and $u_k = u_k^{in}$. It follows

$$\Delta u_k = u_k^{in} - u_{k-1}^{in} = -\bar{B}_k^{-1}(s_k + \bar{e}_{\bar{F}_k} - \bar{e}_{\bar{B}_k}\bar{B}_{k-1}^{-1}\bar{F}_{k-1}) + \bar{B}_{k-1}^{-1}s_{k-1}.$$

Therefore, the control, $\Delta u_k = O(T)$, is continuous as defined in Definition 2. □

III. Matched Disturbances and Prediction

Consider the case when the uncertainties are due to either external disturbances or parameter variations in the dynamic system

$$\dot{x} = f(x) + g(x)u + v(x,t), \tag{15}$$

where $v(x,t)$ is an $n \times 1$ unknown vector-valued differentiable function. The discrete time dynamic equation for the sliding surface vector can be expressed as

$$s_{k+1} = s_k + F_k + B_k u_k + d_k, \tag{16}$$

where

$$d_k = \int_{kT}^{(k+1)T} G(x)v(x,t)dt.$$

We say the uncertainty is "rejectable" if there exists a control u that can eliminate its effect instantaneously in the dynamic system. The uncertainty is said to be "matched" to the control if it lies within the range space of the control coefficient matrix $g(x)$ for $x \in \mathcal{S}$. As seen in (15), the necessary and sufficient condition for $v(x,t)$ to be matched is

$$Rank[g(x), v(x,t)] = Rank[g(x)] \text{ when } x \in \mathcal{S}. \tag{17}$$

It is important that the matching conditions be satisfied for the disturbances to be rejectable in sliding mode. If the closed loop system behavior is independent of $v(x,t)$, we say the controlled system is disturbance invariant.

Lemma 3 *The sampled-data system for (15) with a zero-order-hold at the control input is disturbance invariant only if the continuous time matching condition (17) holds.*

Proof. The sampled-data system system for (15) is

$$x_{k+1} = x_k + \int_{kT}^{(k+1)T} f(x)dt + \int_{kT}^{(k+1)T} g(x)dt\, u_k + \int_{kT}^{(k+1)T} v(x,t)dt. \tag{18}$$

Suppose the system is in sliding mode $s_k = 0$, one can substitute the discrete time equivalent control (obtained by solving $s_{k+1} = 0$ in (16)) into (18) and yield

$$x_{k+1} = x_k + \int_{kT}^{(k+1)T} f(x)dt + \int_{kT}^{(k+1)T} v(x,t)dt - \int_{kT}^{(k+1)T} g(x)dtB_k^{-1}(F_k + d_k). \tag{19}$$

The necessary and sufficient condition for the discrete time system (19) to be disturbance invariant is

$$\int_{kT}^{(k+1)T} g(x)dtB_k^{-1}d_k = \int_{kT}^{(k+1)T} v(x,t)dt,$$

which further implies that there exists a constant $m \times 1$ vector q_k satisfying

$$\int_{kT}^{(k+1)T} v(x,t)dt = \int_{kT}^{(k+1)T} g(x)dtq_k. \tag{20}$$

Therefore, $Rank[g(x), v(x,t)] = Rank[g(x)]$. □

Note that the reverse argument is not true. If the continuous time matching condition holds, it does not guarantee the existence of a constant vector q_k satisfying (20) because (20) is a system of n algebraic equations with m unknowns with $m < n$ in general.

In a sampled system, the control is constrained to be piecewise constant due to the hold process. Since a similar hold does not exist in the disturbance channel, the disturbances are usually not contained in the control signal range space even if the matching condition holds. As a result, the perfect sliding property is destroyed. Our goal is then to find a proper control law that maintains the state in the vicinity of the sliding manifold.

The magnitude of the unknown disturbance d_k is $O(T)$. Note that d_k depends on the future values of the states. Apply the following assumption so that d_k can be predicted by its previous value d_{k-1}.

Assumption 3

1. *The uncertainty* $v(x,t)$ *is bounded and smooth;* $\frac{\partial}{\partial x}[G(x)v(x,t)]$ *is bounded.*
2. $v(x,t)$ *satisfies the continuous time matching condition (17).*
3. *The 2-norm of the admissible control* u_k *is bounded by a positive finite number* M_u *such that* $\|u\| \leq M_u$.

Evaluation of d_{k-1} requires exact values of F_{k-1} and B_{k-1}, which are impossible to evaluate in general. We use the approximated values in (10) and obtain

$$\bar{d}_{k-1} = s_k - s_{k-1} - \bar{F}_{k-1} - \bar{B}_{k-1}u_{k-1}. \tag{21}$$

Adopt the notation:

$$e_{d_k} = d_k - d_{k-1} \quad e_{\bar{d}_k} = d_k - \bar{d}_k \tag{22}$$

It is proved in the Appendix that there exist finite positive numbers M_{d_1}, M_{d_2} that satisfy

$$\|e_{d_k}\| \leq M_{d_1}T^2 \quad \|e_{\bar{d}_k}\| \leq M_{d_2}T^2. \tag{23}$$

Since the magnitude of the error $d_k - \bar{d}_{k-1} = e_{d_k} + e_{\bar{d}_{k-1}}$ is of order $O(T^2)$, by using $\bar{d}_{k-1}$ as the estimate of d_k, the sliding manifold $\mathcal{S}$ can be approached with $O(T^2)$ accuracy. Also, it can be proved that Lemma 1 and Lemma 2 still hold for the following control law:

$$u_k = \begin{cases} -\bar{B}_k^{-1}(s_k + \bar{F}_k + \bar{d}_{k-1}) = u_k^{in} & \text{when } x_k \in \mathcal{S}_T \\ -\bar{B}_k^{-1}(K sgn(s_k) + \bar{F}_k + \bar{d}_{k-1}) = u_k^{out} & \text{otherwise} \end{cases} \tag{24}$$

Theorem 1 *For a nonlinear sampled-data system as given in (15), the effect of the matched uncertainties* $v(x,t)$ *can be reduced to* $O(T^2)$ *if the variable structure control (24) is applied*

Proof. When $x_k \notin \mathcal{S}_T$, u_k^{out} is employed. It will steer the state toward the boundary layer $\mathcal{S}_T$ with sliding surface equation

$$s_{k+1} - s_k = -K sgn(s_k) + e_{\bar{F}_k} + e_{d_k} + e_{\bar{d}_{k-1}},$$

where the positive definite matrix K is chosen small enough so that $u_k^{out} \in \mathcal{U}$.

Once the state enters $\mathcal{S}_T$, u_k^{in} will further maintain $s_{k+1} = O(T^2)$ according to Lemma 1. Thus the effect of the matched disturbance is reduced to $O(T^2)$. □

It is interesting to note that $\bar{F}_k + \bar{d}_{k-1}$ can be further approximated by using the relationship in (21) so that the control (24) can be expressed in an adaptive formulation

$$u_k = \begin{cases} -\bar{B}_k^{-1}(2s_k - s_{k-1} - \bar{B}_{k-1}u_{k-1}) & \text{when } x_k \in \mathcal{S}_T \\ -\bar{B}_k^{-1}(K sgn(s_k) + s_k - s_{k-1} - \bar{B}_{k-1}u_{k-1}) & \text{otherwise,} \end{cases} \tag{25}$$

which does not depend on $\bar{F}_k$ or $f(x)$. In addition, if $\bar{B}_k$ is a constant, the control u_k will be a function of s_k, s_{k-1}, and u_{k-1}.

IV. Control Coefficient Variations

Consider the case when there is an unknown variation $\tilde{g}(x)$ in the control coefficient $g(x)$

$$\dot{x} = f(x) + g(x)u + \tilde{g}(x)u. \tag{26}$$

Systems with uncertainties in the control coefficient matrix have not been studied thoroughly in sliding mode control theory. The difficulty mainly lies in the fact that the induced disturbance is directly related to the control signal itself. However, the difficulty in analysis does not hinder the simplicity in implementation for continuous time sliding mode control. If the variation $\tilde{g}(x)$ is sufficiently small that the sign of each element in the control coefficient $g(x) + \tilde{g}(x)$ coincide with that in $g(x)$, the primary effect would only be to modulate the magnitude of the control signal. Since the control policy is determined by the sliding surface vector $s(x)$, the magnitude of the control coefficient is immaterial as long as it is large enough to dominate the other uncertainties. Therefore, most frequently, the design of continuous time sliding mode control simply ignores $\tilde{g}(x)$.

In discrete time, the same policy can be adopted to steer the state into the $O(T)$ boundary layer of the sliding manifold, $\mathcal{S}_T$. However, the uncertainty needs to be taken into account in control design if we desire a higher accuracy.

Following the same discretization procedure as indicated in (6), we have

$$s_{k+1} = s_k + F_k + B_k u_k + d_k, \tag{27}$$

where

$$\begin{array}{l} d_k = \tilde{B}_k u_k \\ \tilde{B}_k = \int_{kT}^{(k+1)T} G(x)\tilde{g}(x)dt \end{array}$$

The measurable portion of the disturbance is

$$\bar{d}_k = s_{k+1} - s_k - \bar{F}_k - \bar{B}_k u_k.$$

Assumption 4

1. *The unknown function* $\tilde{g}(x)$ *is bounded and smooth;* $\frac{\partial}{\partial x}[G(x)\tilde{g}(x)]$ *is bounded.*
2. $\tilde{g}(x)$ *is matched:* $Rank[g(x), \tilde{g}(x)] = Rank[g(x)]$.
3. *The* $m \times m$ *matrix* $B_k + \tilde{B}_k$ *is nonsingular for all* k.

We propose the control law

$$u_k = \left\{ \begin{array}{ll} -\bar{B}_k^{-1}(s_k + \bar{F}_k + \bar{d}_{k-1}) = u_k^{in} & \text{when } u_k^{in} \in \mathcal{U} \\ -\bar{B}_k^{-1}(K sgn(s_k) + \bar{F}_k + \bar{d}_{k-1}) = u_k^{out} & \text{otherwise} \end{array} \right. \tag{28}$$

It can be observed that the control (28) contains dynamic feedback through the disturbance prediction term $\bar{d}_{k-1}$. According to the relationship in (27), the stability of u_k can be assured if the sliding surface vector s_k is stable. We shall prove that, with a small enough control coefficient matrix variation $\tilde{g}(x)$ and fast enough sampling rate, the control (28) will lead to a stable dynamic equation for s_k and that $s_k \to O(T^2)$ asymptotically.

Introduce the following notation:

$$\begin{array}{ll} e_{F_k} = F_k - F_{k-1}, \quad e_{B_k} = B_k - B_{k-1}, & e_{\tilde{B}_k} = \tilde{B}_k - \tilde{B}_{k-1} \\ \Delta_k = \tilde{B}_k B_k^{-1}, & \epsilon_{\Delta_k} = \Delta_k - \Delta_{k-1}. \end{array} \tag{29}$$

Based on Assumptions 4, the following results can be proved (see Appendix):

There exist finite positive numbers $M_{\tilde{B}}$ and M_Δ such that

$$\begin{array}{lll} \|e_{F_k}\| \le \frac{M_F}{4}T^2, & \|e_{B_k}\| \le \frac{M_B}{4}T^2, & \|e_{\tilde{B}_k}\| \le M_{\tilde{B}}T^2 \\ \|\epsilon_{\Delta_k}\| \le M_\Delta T, & \|I - B_k\bar{B}_k^{-1}\| \le \frac{M_B}{m_g}T. & \end{array} \tag{30}$$

Lemma 4 *When the state is in the boundary layer $\bar{\mathcal{S}}$, the following equalities are true:*

1.

$$s_{k+1} = d_k - d_{k-1} + \alpha_k^1(2s_k - s_{k-1}) + \alpha_k^2, \tag{31}$$

where $\alpha_k^1 = O(T)$, $\alpha_k^2 = O(T^2)$.

2.

$$d_k = -\Delta_k[(s_k + F_k + d_{k-1}) + \beta_k^1(2s_k - s_{k-1}) + \beta_k^2], \tag{32}$$

where $\beta_k^1 = O(T)$, $\beta_k^2 = O(T^2)$.

The equalities in Lemma 4 can be obtained by direct substitution from (22) and (29). It yields

$$\begin{aligned} \alpha_k^1 &= -2e_{\bar{B}_k}\bar{B}_k^{-1} \\ \alpha_k^2 &= -e_{\bar{B}_k}B_k^{-1}F_k + e_{\bar{F}_k} + e_{\bar{B}_k}\bar{B}_k^{-1}e_{\bar{F}_k} + e_{\bar{d}_{k-1}} + e_{\bar{B}_k}\bar{B}_k^{-1}(\bar{F}_{k-1} + \bar{B}_{k-1}u_{k-1}) \\ \beta_k^1 &= e_{\bar{B}_k}B_k^{-1} \\ \beta_k^2 &= -e_{\bar{F}_k} - e_{\bar{d}_{k-1}}e_{\bar{B}_k}\bar{B}_k^{-1}(e_{\bar{F}_k} - \bar{B}_{k-1}u_{k-1}) \end{aligned} \tag{33}$$

They serve as preparation for the proof of the theorem stated below:

Theorem 2 *The control in (28) will lead to*

$$s_{k+1} = a_k^0 s_k + a_k^1 s_{k-1} + a_k^2 s_{k-2} + \delta_k, \tag{34}$$

where

$$\begin{aligned} a_k^0 &= -2\Delta_k + O(T) \\ a_k^1 &= \Delta_{k-1} + O(T) \\ a_k^2 &= O(T) \\ \delta_k &= O(T^2) \end{aligned}$$

Proof. If $x_k \notin \mathcal{S}_T$, u_k^{out} will steer the state toward the boundary layer $\mathcal{S}_T$. When $x_k \in \mathcal{S}_T$ Substituting (32) into (31) yields

$$\begin{aligned} a_k^0 &= -2\Delta_k - 2\Delta_k\beta_k^1 + 2\alpha_k^1 \\ a_k^1 &= \Delta_{k-1} + 2\Delta_k\alpha_k^1 - \epsilon_{\Delta_k} + \Delta_k\beta_k^1 + 2\Delta_{k-1}\beta_k^1 - \alpha_k^1 \\ a_k^2 &= \epsilon_{\Delta_k} - \Delta_k\alpha_k^1 - \Delta_{k-1}\beta_k^1 \\ \delta_k &= -\Delta_k e_{F_k} - \epsilon_{\Delta_k}F_{k-1} + \Delta_k\alpha_k^2 + \epsilon_{\Delta_k}(F_{k-2} + B_{k-2}u_{k-2}) - \Delta_k\beta_k^2 + \Delta_{k-1}\beta_{k-1}^2 + \alpha_k^2 \end{aligned}$$

□

If the sampling rate is sufficiently high, the dynamic equation (34) is reduced to second order

$$s_{k+1} = -2\Delta_k s_k + \Delta_{k-1}s_{k-1}. \tag{35}$$

Stability of s_k can be assured if Δ_k is small enough; that is, the variation of the control coefficient $\tilde{g}(x)$ is small with respect to $g(x)$. This result justifies the ignoring of $\tilde{g}(x)$ in continuous time variable structure control design. If the $O(T)$ terms in (34) are not negligible, the dynamic equation for s_k becomes third order with an $O(T^2)$ input. Stability of (34) implies that $s_k \to O(T^2)$ asymptotically.

V. Conclusion

In this paper, we have investigated the sample/hold effect and taken it into account for discrete time sliding mode control design. We proved that the piecewise constant control law due to zero-order-hold has an inherent limitation in disturbance rejection. Even if the sampled-data system is kept exactly in the sliding manifold at each sampling instant, the intersample behavior still deviates from $\mathcal{S}$ by the amount $O(T^2)$. The error is inevitable since a similar "hold" does not take place in the disturbance channels. The sample/hold effect invalidates one's attempt to achieve complete disturbance invariance. Nevertheless, an $O(T^2)$ accuracy is sufficient in most applications; after all, there is no perfect sliding mode in reality.

A key assumption throughout the derivation is boundedness and smoothness of the nonlinearities and disturbances. It allows one to approximate integrations and predict future unknowns by using past, discrete measurement. The induced error leads to an $O(T^2)$ boundary layer in the neighborhood of the sliding manifold.

The idea of boundary layer enables one to smooth the control instead of instantaneous switchings. It is superior to the saturation function approach, which compromises chattering with robustness [8]. The proposed method is is simple from an implementation point of view. It deals with both exogenous disturbances and internal uncertainties. Furthermore, the notorious chattering problem associated with the continuous time variable structure control is removed with considerable robustness.

References

[1] V. I. Utkin, *Sliding modes and their application in variable structure systems.* Moscow: MIR, 1978.

[2] V. Utkin and S. Drakunov, "On discrete-time sliding modes," in *Preprints of IFAC Workshop on Nonlinear Control*, (Capri, Italy), 1989.

[3] C. Milosavljevic, "General conditions for the existence of a quasisliding mode on the swithing hyperplane in discrete variable," *Automat. Remote Contr.*, vol. 46, pp. 307–314, 1985.

[4] S. Sarptürk, Y. Istefanopulos, and O. Kaynak, "On the stability of discrete-time sliding mode control systems," *IEEE Trans. Automat. Contr.*, vol. AC-32, no. 10, pp. 930–932, October 1987.

[5] K. Furuta, "Sliding mode control of a discrete system," *Systems and Control Letters*, vol. 14, pp. 145–152, 1990.

[6] S. Drakunov and V. Utkin, "Sliding mode control in dynamic systems," *Int. J. Control*, vol. 55, no. 4, pp. 1029–1037, 1992.

[7] W. C. Su, S. V. Drakunov, and Ü. Özgüner, "Sliding mode contol in discrete time linear systems," in *Proceedings of the 12th IFAC World Congress*, (Sidney, Australia), July 1993.

[8] K. Young and S. Drakunov, "Sliding mode control with chattering reduction," in *Proceedings of the 1992 American Control Conference*, (Chicago, IL), pp. 1291–1292, June 1992.

Appendix

Proof of (12),(23), and (30):

1.

$$e_{\bar{F}_k} = \int_{kT}^{(k+1)T} G(x)f(x)d\tau - G(x_k)f(x_k)T$$

Let $e_{\bar{F}_k}(t) = \int_{kT}^{kT+t} G(x)f(x)d\tau - G(x_k)f(x_k)t$, $0 \le t \le T$, then

$$\ddot{e}_{\bar{F}_k}(t) = \frac{\partial}{\partial x}[G(x)f(x)][f(x) + g(x)u_k]|_{x=x(kT+t)}$$

With Assumption 2, $\exists\ M_F$ such that $|\ddot{e}_{\bar{F}_k}(t)| \le \frac{M_F}{2}$. It yields,

$$|e_{\bar{F}_k}| \le M_F T^2.$$

□

2.

$$\bar{e}_{\bar{F}_k} = T \int_0^T \frac{\partial}{\partial \tau}[G(x(kT - T + \tau))f(x(kT - T + \tau))]d\tau$$

$$\rightarrow \quad \|e_{\bar{F}_k}\| \le \frac{M_F}{2}T^2.$$

□

3.

$$e_{F_k} = \int_0^T \int_0^T \frac{\partial}{\partial \tau}[G(x(kT - T + \tau))f(x(kT - T + \tau))]d\tau$$

$$\rightarrow \quad \|e_{\bar{F}_k}\| \le \frac{M_F}{4}T^2.$$

□

4.

$$e_{\bar{d}_k} = -e_{\bar{F}_k} - e_{\bar{B}_k}u_k$$

With Assumption 3, u_k is bounded, therefore, $\exists\ M_{d_2}$ such that $\|e_{\bar{d}_k}\| \le M_{d_2}T^2$. □

5.

$$\epsilon_{\Delta_k} = \tilde{B}_k B_k^{-1} - \tilde{B}_{k-1} B_{k-1}^{-1} = e_{\tilde{B}_k} B_k^{-1} - \tilde{B}_{k-1} B_{k-1}^{-1} e_{B_k} B_k^{-1}.$$

$$\rightarrow \quad \|\epsilon_{\Delta_k}\| \frac{M_{\tilde{B}}}{m_g}T + \frac{M_{\tilde{B}} M_B}{4m_g^2}T = M_\Delta T.$$

□

6.

$$I - B_k \bar{B}_k^{-1} = -e_{\bar{B}_k} \bar{B}_k^{-1}$$

$$\rightarrow \quad \|I - B_k \bar{B}_k^{-1}\| \le \frac{M_B}{m_g}T.$$

□

7. similar proof for e_{B_k}, $e_{\tilde{B}_k}$, $\bar{e}_{\bar{B}_k}$, and e_{d_k}.

Dynamical Systems Approach to Vision Based Ranging from Feature Observations

Mrdjan Jankovic and Bijoy K. Ghosh *
Department of Systems Science and Mathematics
Washington University, Campus Box 1040
St. Louis, Missouri 63130

Abstract

In this paper we consider the problem of estimating the range information of features on an affine plane in $\mathbb{R}^3$ by observing its image with the aid of a CCD camera, wherein we assume that the camera is undergoing a known motion. The features considered are points, lines and curves located on planar surfaces of static objects. The dynamics of the moving projections of the features on the image plane have been described as a suitable differential equation on an appropriate feature space. This dynamics is used to estimate feature parameters from which the range information is readily available. In this paper the proposed identification has been carried out via a newly introduced Identifier Based Observer.

1 Introduction

In this paper we consider the problem of range estimation from a sequence of images generated by a moving camera. We assume that the camera rotates and translates with known, possibly time varying, angular and linear velocities respectively. The camera may be mounted on a rotorcraft, an autonomous vehicle, a robot (in the so called 'eye in hand' configuration) or a moving platform designed solely for the purpose of range estimation.

We assume that we have identified a suitable feature on the image plane and that we are able to establish a correspondence of this feature between consecutive frames. The features considered in this paper are points, lines, and planar curves. The main result that we show in this paper is that by tracking the motion of a feature on the image plane, one is able to find the true position of the feature in the 3D space.

The above problem of estimating the location of features can be posed as a problem of state estimation of a nonlinear system. In order to do this we first establish that the features (viz. points, lines and curves) can be parameterized as points in a feature space which can be given the structure of an analytic manifold.

Once an appropriate parameterization has been obtained, we need to describe the dynamics of the feature as a consequence of the motion of the camera. Feature Dynamics is described as a dynamical system on the feature space with state variables describing the location of the features.

Finally, we need to quantify a set of measured (observed) variables, i.e. variables that can be recovered from a sequence of images. In order to design an estimator, states of the dynamical system must be *observable* from the measured variables in order for the estimates to converge to the true values of parameters.

---

*Partially supported by DOE under grant No. DE-FG02-90ER14140.

The problem of range estimation, when the observed features are points, has already been considered in a number of papers including [1], [6],[7], [9], [10], [11]. In all of these references except [7] the Extended Kalman Filter (EKF) has been used. The case of range estimation from observation of lines as features has been considered in [1] and [2] from the images from stationary multi-camera system. As far as we know, the problem of estimation of range data from an observation of planar curves, is new.

The dynamics for all the tree different features considered here have turned out to belong to the same class of nonlinear systems. For this class of systems we have developed a state estimator which we have called Identifier Based Observer. The EKF can be used as well, but for the Identifier Based Observer we have obtained sufficient conditions for convergence which can be monitored on-line.

2 Identifier Based Observer

In this section we develop an observer for nonlinear systems based on a parameter identifier. A class of nonlinear systems for which this observer is applicable is given by

$$\begin{aligned} \dot{x}_1 &= w^T(x_1,u)x_2 + \phi(x_1,u) \\ \dot{x}_2 &= g(x_1,x_2,u) \\ y &= x_1 \end{aligned} \tag{2.1}$$

where $x_1 \in X_1 \subset R^{n_1}, x_2 \in X_2 \subset R^{n_2}$ and $u \in U \subset R^k$. Denote by $x = [x_1, x_2]^T$. It follows that for $n = n_1+n_2$, $x \in X \subset R^n$ and $X = X_1 \oplus X_2$. The $n_1 \times n_2$ matrix $w^T(x_1,u)$ and the vector $g(x_1,x_2,u)$ are in general nonlinear functions of their arguments.

In order to design the observer we need the nonlinear vector g to satisfy Lipschitz condition. Since the global Lipschitz condition is too restrictive we shall assume boundedness of the states of the system (2.1) and the local Lipshitz condition. If the global Lipschitz condition is satisfied we don't need the states to be bounded.

Assumption 2.1: *Suppose that there exists a constant $M > 0$ such that $\|x(t)\| < M$ for every $t > 0$. Denote by Ω the following set; $\Omega = \{x \in R^n : \|x(t)\| < M\}$. Assume that the function $g(x_1,x_2,u)$ satisfies Lipschitz condition in Ω with respect to x_2, i.e. assume that there exists a positive constant α, such that*

$$\|g(x_1,x_2,u) - g(x_1,z_2,u)\| < \alpha\|x_2 - z_2\| \tag{2.2}$$

for all $x_2, z_2 \in \Omega \bigcap X_2$, uniformly in $x_1 \in \Omega \bigcap X_1$ and $u \in U$.

The second assumption is an observability assumption. It resembles the persistence of excitation condition, (see, for example, [8] page 247) but is stronger.

Assumption 2.2: *Suppose that the regressor matrix $w^T(x_1,u)$ is piecewise smooth and suppose that there exist positive constants L_1, L_2, r, and β such that*

$$\|w^T(x_1,u)\| < L_1 \text{ and } \|\frac{dw^T(x_1,u)}{dt}\| < L_2 \tag{2.3}$$

$$\rho\ \beta I < \int_t^{t+\rho} w(x_1(\tau),u(\tau))\ w^T(x_1(\tau),u(\tau))\ d\tau \tag{2.4}$$

for all $0 < \rho < r$ for all $t \in \mathbb{R}^+$, for all trajectories that originate in X and for all applied inputs that are in U.

While the above observability assumption (2.4) is given in an integral form in order to simplify the proof of convergence of the estimation algorithm, it is not difficult to see that (2.4) is equivalent to the assumption that for some positive ε_1

$$\lambda_{min}\{w(x_1(t),u(t))w^T(x_1(t),u(t))\} > \varepsilon_1 \tag{2.5}$$

for all $t \in \mathbb{R}^+$ where λ_{min} denotes the smallest eigenvalue.

Let us now consider the following observer for the nonlinear system (2.1).

$$\begin{aligned} \dot{\hat{x}}_1 &= GA(\hat{x}_1 - x_1) + w^T(x_1, u)\hat{x}_2 + \phi(x_1, u) \\ \dot{\hat{x}}_2 &= -G^2 w(x_1, u)P(\hat{x}_1 - x) + g(x_1, \bar{x}_2, u) \\ \bar{x}(t) &= M \text{ sat}\left(\frac{\|\hat{x}(t)\|}{M}\right)\frac{\hat{x}(t)}{\|\hat{x}(t)\|} \end{aligned} \tag{2.6}$$

where sat$(\cdot)$ is the saturation function, $\bar{x} \triangleq [\bar{x}_1 \; \bar{x}_2]^T$, and where A is an $n \times n$ Hurwitz matrix. The matrix P is a positive definite solution of the Liapunov matrix equation $A^T P + PA = -Q$ with Q a positive definite symmetric matrix. We shall refer to the above observer (2.6) as the Identifier Based Observer.

If we define estimation errors e_1 and e_2 by $e_1 \triangleq \hat{x}_1 - x_1$ and $e_2 \triangleq \hat{x}_2 - x_2$ the error dynamics takes the following form:

$$\begin{aligned} \dot{e}_1 &= GAe_1 + w^T(x_1, u)e_2 \\ \dot{e}_2 &= -G^2 w(x_1, u)Pe_1 + g(x_1, x_2, u) - g(x_1, \bar{x}_2, u) \end{aligned} \tag{2.7}$$

Theorem 2.3: *If the assumptions 2.1 and 2.2 are satisfied then there exists a positive constant G_0 such that the estimation errors e_1 and e_2 converge to zero exponentially if the constant G in (2.6) is chosen larger than G_0.*

The proof can be found in [4].

3 Feature Based Range Estimation

In this section we consider the problem of range estimation by tracking images of moving features on the screen. The features that we consider are points, lines and planar curves. We propose appropriate parameterizations of the space of features. As a result of the camera motion the features move and the motion is described by a dynamical system on the parameter space. The dynamical systems have been obtained for all the three different features. Finally, these dynamics have also been used to design estimators for the unknown part of the parameter vector.

3.1 Range Estimation from Feature Points

First we consider the problem of range estimation from the observation of points. We assume that a reasonably accurate correspondence between points in consecutive images has been established. For further discussion on the problem of correspondence we refer to [3], [6], and [9].

First problem in range estimation is to find an appropriate model which connects the location of the features (in this case points) on the screen and the variables we want to estimate (in this case range). Let $\omega_1, \omega_2, \omega_3$ and v_1, v_2, v_3 denote respectively the components of the known angular and linear velocities in the coordinate system attached to the camera. We assume that all the velocities and accelerations are bounded.

It is well known that the coordinates X, Y, Z of a stationary point P in the moving camera coordinate frame satisfy

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} = \begin{bmatrix} 0 & \omega_3 & -\omega_2 \\ -\omega_3 & 0 & \omega_1 \\ \omega_2 & -\omega_1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} - \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} \triangleq \Omega^T \mathcal{X} - v \tag{3.1}$$

with the obvious meaning of the symbols Ω, $\mathcal{X}$, and v.

By accepting a pinhole model of the camera the coordinates of the image of the point P are given by $x = f\frac{X}{Z}$ and $y = f\frac{Y}{Z}$ where f stands for the focal length. To simplify the notation from now on we shall assume that $f = 1$. Note that the variables x and y represent our measurements and the dynamics (3.1) can be expressed in x, y and $d \triangleq 1/Z$ coordinates as follows:

$$\begin{aligned} \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} &= \begin{bmatrix} -v_1 + xv_3 \\ -v_2 + yv_3 \end{bmatrix} d + \begin{bmatrix} xy\omega_1 - (1+x^2)\omega_2 + y\omega_3 \\ (1+y^2)\omega_1 - xy\omega_2 - x\omega_3 \end{bmatrix} \\ \dot{d} &= (\omega_1 y - \omega_2 x)d + v_3 d^2 \end{aligned} \tag{3.2}$$

A quick comparison of the above model (3.2) with the class of systems described by (2.1) suggests that the Identifier Based Observer might be applicable to estimate the range (or rather the inverse of the range) of the point.

In order to guarantee the convergence of the estimation errors to 0 we must assure that Assumptions 2.1 and 2.2 are satisfied. For the problem in hand it is reasonable to assume that the states of the system are bounded. First of all, in order for the point to appear on the screen, x and y must be bounded. Moreover the variable d , the inverse of the depth Z, is positive and must be smaller than $1/f$ where f is the focal length of the camera (in this case $f = 1$). Therefore, Assumption 2.1 is satisfied.

According to the observability condition (2.5) the system (3.2) is observable, and the Assumption 2.2 is satisfied, if the 2×1 matrix multiplying d is of uniform rank 1, or equivalently if

$$(v_1 - xv_3)^2 + (v_2 - yv_3)^2 > \varepsilon^2$$

for some $\varepsilon > 0$. For $v_3 \neq 0$ the above expression defines a complement of a circle on the screen with the center at $(v_1/v_3, v_2/v_3)$ and the radius ε/v_3. The point $(v_1/v_3, v_2/v_3)$ is called the *Focus of Expansion* (FOE). It is a well known fact that the distance of a feature point at the FOE cannot be determined. In addition, in the presence of measurement noise the distance of points close to the FOE cannot be determined reliably. Therefore, the above sufficient observability condition is, for practical purposes, also necessary.

3.2 Range Estimation from Observing Straight Lines

In this subsection we turn to the problem of range estimation from a sequence of projections of a line. The motivation for this comes from the fact that it is much more reliable to establish correspondences between lines in two consecutive images than it is for points.

In order to estimate the equation of the line in the 3-D space we need a parameterization of lines in $\mathbb{R}^3$. Such a parameterization problem has been already considered in [1] and [2]. The parameterization obtained therein is however different from the one proposed here. Additionally in [1] and [2] the parameter estimation problem is based upon images from multiple cameras. In order to use the IBO machinery proposed in section 2, we shall obtain a parameterization of the set of affine lines in $\mathbb{R}^3$ as points in $Grass(2,4)$ the Grassmannian manifold of homogeneous 2 *dimensional* planes in $\mathbb{R}^4$. This parameterization of affine lines has been proposed in [5] in order to estimate the motion parameters of a moving object with the aid of a single camera. For the sake of completeness we shall repeat some of the basic results already outlined in [5].

We start with the parameterization of an affine line in $\mathbb{R}^3$ given as the intersection of two planes, i.e. the line is given by the set of points in $\mathbb{R}^3$ which satisfy

$$m_1X + m_2Y + m_3Z + m_4 = 0$$
$$m_5X + m_6Y + m_7Z + m_8 = 0$$

where obviously vectors $[m_1\ m_2\ m_3\ m_4]^T$ and $[m_5\ m_6\ m_7\ m_8]^T$ must be linearly independent. From the above parameterization of a line we define six Plücker coordinates as follows.

$$\eta_{14} = \det\begin{bmatrix} m_1 & m_4 \\ m_5 & m_8 \end{bmatrix} \quad \eta_{24} = \det\begin{bmatrix} m_2 & m_4 \\ m_6 & m_8 \end{bmatrix} \quad \eta_{34} = \det\begin{bmatrix} m_3 & m_4 \\ m_7 & m_8 \end{bmatrix}$$

$$\eta_{12} = \det\begin{bmatrix} m_1 & m_2 \\ m_5 & m_6 \end{bmatrix} \quad \eta_{23} = \det\begin{bmatrix} m_2 & m_3 \\ m_6 & m_7 \end{bmatrix} \quad \eta_{31} = \det\begin{bmatrix} m_3 & m_1 \\ m_7 & m_5 \end{bmatrix}$$

It is not difficult to show that the variables η_{ij} satisfy $\eta_{12}^2 + \eta_{23}^2 + \eta_{31}^2 \neq 0$ (see [5]). Let η denote the vector of Plücker coordinates. Since η is defined up to a scale factor, one can normalize η as follows.

$$\begin{bmatrix} u_1 & u_2 & u_3 & u_4 & u_5 & u_6 \end{bmatrix}^T = \frac{1}{\sqrt{\eta_{12}^2 + \eta_{23}^2 + \eta_{31}^2}} \begin{bmatrix} \eta_{14} & \eta_{24} & \eta_{34} & \eta_{12} & \eta_{23} & \eta_{31} \end{bmatrix}^T$$

where $u_j, j = 1 \cdots 6$ is a set of normalized Plücker coordinates. For each affine line in $\mathbb{R}^3$ there is a corresponding vector of 6 Plücker coordinates. The converse is however not true. For a vector of Plücker coordinates to corespond to a line, it must satisfy

$$u_1u_4 + u_2u_5 + u_3u_6 = 0 \tag{3.3}$$

According to this expression two vectors of dimension 3 defined via $\bar{u}_1 = [u_1\ u_2\ u_3]$ and $\bar{u}_2 = [u_4\ u_5\ u_6]$ are orthogonal. In fact these two vectors have a clear geometric meaning. The vector $\bar{u}_2$ is a unit vector in the direction of the line while the vector $\bar{u}_1$ is the cross-product between $\bar{u}_2$ and the vector from the origin to the closest point on the line.

Even though the parameterization of affine lines via Plücker coordinates is an over-parameterization (recall that the set of affine lines in $\mathbb{R}^3$ is a manifold of dimension 4, see [2], whereas the six Plücker coordinates defined homogeneously describe a projective space of dimension 5), such a parameterization is quite useful. This is because the dynamics satisfied by the vector u can be easily obtained from the infinitesimal version of the result given in Theorem 1 [5] and is given by:

$$\begin{bmatrix} \dot{\bar{u}}_1 \\ \dot{\bar{u}}_2 \end{bmatrix} = \begin{bmatrix} \Omega^T & V \\ 0 & \Omega^T \end{bmatrix} \begin{bmatrix} \bar{u}_1 \\ \bar{u}_2 \end{bmatrix} \tag{3.4}$$

where the matrix V is defined by

$$V = \begin{bmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{bmatrix}$$

We now claim that among the Plücker coordinates $u_1 \cdots u_6$ it is possible to measure u_1, u_2, u_3 upto a scale factor from the observation of projections of affine lines. Note that it has been already shown in [5] that the entries of the vector $\bar{u}_1$ satisfy

$$u_1x + u_2y + u_3 = 0 \tag{3.5}$$

Obviously, from the equation of the line on the image plane the vector $\bar{u}_1$ can be obtained up to a scale factor.

Now in order to be able to design an observer we shall have to reparameterize the system of equation (3.4) such that the lines are parameterized with the minimal number of parameters, in this case 4. However, since the manifold of lines is not diffeomorphic to $\mathbb{R}^4$ every parameterization is valid only on a subset of the set of lines. Therefore, assume without any loss of generality that $u_3 \neq 0$. The situation is analogous if either $u_1 \neq 0$ or $u_2 \neq 0$. Geometrically $u_3 \neq 0$ implies that the projection of the line on the image plane does not pass through the origin of the image plane. Under this assumption we can divide the equation (3.5) by u_3 to obtain $b_1 x + b_2 y + 1 = 0$ where obviously $b_1 = u_1/u_3$ and $b_2 = u_2/u_3$.

Define the variables p_1, p_2, and p_3 via $p_1 = u_4/u_3$, $p_2 = u_5/u_3$, $p_3 = u_6/u_3$. Then the following proposition gives the dynamics for the minimal set of line parameters b_i, $i = 1,2$ and p_i, $i = 1,2$.

Proposition 3.1: *If the line (3.5) does not pass through the origin of the image plane, then the dynamics for the 4 parameters b_i, $i = 1,2$ and p_i, $i = 1,2$ is given by*

$$\begin{bmatrix} \dot{b}_1 \\ \dot{b}_2 \end{bmatrix} = \begin{bmatrix} 0 & -b_1 v_1 - b_2 v_2 - v_3 \\ b_1 v_1 + b_2 v_2 + v_3 & 0 \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \end{bmatrix} + \begin{bmatrix} b_1 b_2 \omega_1 - (1 + b_1^2)\omega_2 + \omega_3 b_2 \\ (1 + b_2^2)\omega_1 - b_1 b_2 \omega_2 - b_1 \omega_3 \end{bmatrix}$$
$$\begin{bmatrix} \dot{p}_1 \\ \dot{p}_2 \end{bmatrix} = \begin{bmatrix} b_1 \omega_1 & \omega_3 + b_2 \omega_2 \\ -\omega_3 - b_1 \omega_1 & -b_2 \omega_2 \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \end{bmatrix} + \begin{bmatrix} v_2 p_1^2 - v_1 p_1 p_2 \\ v_2 p_1 p_2 - v_1 p_2^2 \end{bmatrix} \tag{3.6}$$

The proof follows by direct calculation of the derivatives of b and p using their definitions and the derivatives of u_i's given by (3.4). The dependence of the equations (3.6) on p_3 is avoided by substituting $p_3 = -b_1 p_1 - b_2 p_2$ which follows directly from (3.3) and the definitions of p and b. The observation (measurement) vector (b_1, b_2) is available from the equation of the line on the image plane given by: $b_1 x + b_2 y + 1 = 0$.

Note that the system (3.6) is in the form for which the Identifier Based Observer is applicable. In order to guarantee that the identifier indeed converges, we must verify that the two assumptions are satisfied. Assumption 2.1 is satisfied if the image of the line on the screen does not pass through a small neighborhood of the origin on the screen. Assumption 2.2, on the other hand, is satisfied if the components of the linear velocity are bounded and

$$|b_1 v_1 + b_2 v_2 + v_3| > \varepsilon$$

for some positive constant ε. This condition is in agreement with the intuition. When $|b_1 v_1 + b_2 v_2 + v_3| = 0$ it follows that the velocity vector, the origin in $\mathbb{R}^3$ and the line are in the same plane; i.e. the camera is moving directly towards the line being observed. If this is the case the image of the line does not move on the screen and obviously the parameters p cannot be estimated.

Once we have the estimates of b and p we can solve back for Plücker coordinates using $p_3 = -b_1 p_1 - b_2 p_2$, $u_3 = \frac{1}{\sqrt{p_1^2 + p_2^2 + p_3^2}}$, $u_1 = b_1 u_3$, $u_2 = b_2 u_3$, $u_4 = u_3 p_1$, $u_5 = u_3 p_2$, $u_6 = u_3 p_3$.

3.3 Range Estimation from Observing Planar Curves

Similar to what has been done in the previous subsection, in this subsection we estimate the equation of a planar curve in $\mathbb{R}^3$ from a sequence of projections on the image plane obtained from a moving camera. Our starting point is a very general parameterization of a second order planar curve in 3-D space given by the intersection of a plane and a second order surface described as follows:

$$\begin{aligned} \zeta_1 X^2 + \zeta_2 Y^2 + \zeta_3 Z^2 + \zeta_4 XY + \zeta_5 XZ + \zeta_6 YZ + \zeta_7 X + \zeta_8 Y + \zeta_9 Z + \zeta_{10} &= 0 \\ \alpha_1 X + \alpha_2 Y + \alpha_3 Z + \alpha_4 &= 0 \end{aligned} \tag{3.7}$$

It is not difficult to see that the above equation for a planar curve in the $\mathbb{R}^3$ is overparameterized. In order to obtain a minimal parameterization we proceed as follows.

Assume that the plane in which the second order curve lies, does not pass through the origin of $\mathbb{R}^3$. By dividing the above equation (3.7) by Z and substituting the expression for $1/Z$ from the second equation into the first, we obtain the following two equations that describe a planar second order curve:

$$\begin{aligned} &\eta_1 x^2 + \eta_2 y^2 + \eta_3 xy + \eta_4 x + \eta_5 y + \eta_6 = 0 \\ &a_1 x + a_2 y + a_3 + \tfrac{1}{Z} = 0 \end{aligned} \tag{3.8}$$

where, as before, $x = X/Z$, $y = Y/Z$, $a_i = \alpha_i/\alpha_4$, $i = 1, 2, 3$ and η's are appropriate expressions involving ζ's and α's. We are allowed to divide by α_4 because, by our assumption that the plane doesn't go through the origin, $\alpha_4 \neq 0$.

Few remarks are now in order. First, the first equation in (3.8) is exactly the equation of the projection of the second order curve on the image plane. Therefore the parameter vector η is observed (measured) up to a scale factor Second, there are 9 parameters in (3.8) describing a planar curve in $\mathbb{R}^3$. However, 6 parameters of the first equation are unique only up to a scale factor which means that one of them can be chosen arbitrarily. This leaves us with a minimal parameterization with 8 parameters. Third, not all combination of parameters in (3.8) define a valid second order curve. For example, for $\eta = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$ there are no points in $\mathbb{R}^3$ satisfying the first equation in (3.8).

Assuming that every point in $\mathbb{R}^3$ satisfies the differential equation (3.1) the dynamical systems for η and a are derived. The result is stated in the following proposition.

Proposition 3.2: *If as a result of camera motion, points in $\mathbb{R}^3$ satisfy the differential equation described by (3.1) then parameters η and a of a planar second order curve satisfy the following differential equations:*

$$\begin{aligned} \dot{\eta} &= \begin{bmatrix} A & 0 & 0 \\ 0 & B & 0 \\ B & A & 0 \\ C & 0 & A \\ 0 & C & B \\ 0 & 0 & C \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} + c\eta + \begin{bmatrix} \eta_3\omega_3 - \eta_4\omega_2 \\ -\eta_3\omega_3 + \eta_5\omega_1 \\ -2\eta_1\omega_3 + 2\eta_2\omega_3 + \eta_4\omega_1 - \eta_5\omega_2 \\ 2\eta_1\omega_2 - \eta_3\omega_1 + \eta_5\omega_3 - 2\eta_6\omega_2 \\ -2\eta_2\omega_1 + \eta_3\omega_2 - \eta_4\omega_3 + 2\eta_6\omega_1 \\ \eta_4\omega_2 - \eta_5\omega_1 \end{bmatrix} \\ &\doteq \Theta(\eta, v)a + c\eta + \phi(\eta, \omega) \end{aligned} \tag{3.9}$$

where $A = -2v_1\eta_1 - v_2\eta_2 - v_3\eta_3$, $B = -v_1\eta_3 - 2v_2\eta_2 - v_3\eta_4$, $C = -v_1\eta_4 - v_2\eta_5 - 2v_3\eta_6$, c is an arbitrary constant and

$$\dot{a} = \Omega^T a - (a^T v)a \tag{3.10}$$

Proof: We begin by deriving the differential equation for the variables in the vector η. As the camera moves, the coordinates $[X\ Y\ Z]^T$ of points in $\mathbb{R}^3$ satisfy (3.1). By (3.8) at each point on the image plane which belongs to the image of the curve the following must be satisfied:

$$\begin{bmatrix} x^2 & y^2 & xy & x & y & 1 \end{bmatrix} \eta(t) = 0 \quad \forall t \geq 0$$

After multiplying the above equation by Z, taking a transpose, differentiating, and finally dividing the resulting equality by Z (in this order) we obtain the following equation:

$$\dot{\eta}^T \begin{bmatrix} x^2 \\ y^2 \\ xy \\ x \\ y \\ 1 \end{bmatrix} + \eta^T \begin{bmatrix} 2xy\omega_3 - 2x\omega_2 - 2v_1 x\frac{1}{Z} \\ -2xy\omega_3 + 2y\omega_1 - 2v_2 y\frac{1}{Z} \\ -x^2\omega_3 + x\omega_1 + y^2\omega_3 - y\omega_2 - xv_2\frac{1}{Z} - yv_1\frac{1}{Z} \\ y\omega_3 - \omega_2 + x^2\omega_2 - xy\omega_1 - v_1\frac{1}{Z} + xv_3\frac{1}{Z} \\ -x\omega_3 + \omega_1 + xy\omega_2 - y^2\omega_1 - v_2\frac{1}{Z} - yv_3\frac{1}{Z} \\ 2x\omega_2 - 2y\omega_1 - 2v_3\frac{1}{Z} \end{bmatrix} = 0 \tag{3.11}$$

We can now solve the second equation in (3.8) for $\frac{1}{Z}$ to obtain $\frac{1}{Z} = -a_1x - a_2y - a_3$. By substituting the right hand side for $\frac{1}{Z}$ in (3.11) we obtain

$$(\dot{\eta}^T + \eta^T\Theta(\omega, v, a)) \begin{bmatrix} x^2 & y^2 & xy & x & y & 1 \end{bmatrix}^T = 0 \tag{3.12}$$

with the matrix $\Theta(\omega, v, a)$ given by

$$\Theta = \begin{bmatrix} 2v_1a_1 & 0 & 2v_2a_2 + 2\omega_3 & 2v_1a_3 - 2\omega_2 & 0 & 0 \\ 0 & 2v_2a_2 & 2v_2a_1 - 2\omega_3 & 0 & 2v_2a_3 + 2\omega_1 & 0 \\ v_1a_1 - \omega_3 & v_1a_2 + \omega_3 & v_2a_2 + v_1a_1 & v_2a_3 + \omega_1 & v_1a_3 - \omega_2 & 0 \\ v_3a_1 + \omega_2 & 0 & v_3a_2 - \omega_1 & v_1a_1 + v_3a_3 & v_1a_2 + \omega_3 & v_1a_3 - \omega_2 \\ 0 & v_3a_2 - \omega_1 & v_3a_1 + \omega_2 & v_2a_1 - \omega_3 & v_2a_2 + v_3a_3 & v_2a_3 + \omega_1 \\ 0 & 0 & 0 & 2v_3a_1 + 2\omega_2 & 2v_3a_2 - 2\omega_1 & 2v_3a_3 \end{bmatrix}$$

The equation (3.12) must be satisfied at each point (x, y) that belongs to the projection of the curve on the image plane. By selecting different points (x, y) in $\mathbb{R}^2$ we obtain vectors $[x^2\ y^2\ xy\ x\ y\ 1]$ which span the six dimensional space. However, if we restrict points to belong to the image of the curve on the image plane the vectors $[x^2\ y^2\ xy\ x\ y\ 1]$ span a five dimensional subspace defined by $\{r \in \mathbb{R}^6 : \eta^T r = 0\}$. Since (3.12) is satisfied at every point on the projection of the curve we conclude that η must satisfy the following differential equation:

$$\dot{\eta}^T + \eta^T\Theta(\omega, v, a) = c\eta^T$$

for some arbitrary constant c.

The last equation together with (3.12) lead to the expression (3.9), after some elementary manipulation, which finishes the first part of the proof.

In order to obtain the dynamical model for the parameters "a" of the plane we return to the expression of the plane given by

$$\alpha_1X + \alpha_2Y + \alpha_3Z + \alpha_4 \triangleq \begin{bmatrix} X & Y & Z & 1 \end{bmatrix}^T \alpha = 0$$

Since this condition must be satisfied for all t and for all points in the plane it follows that

$$\frac{d}{dt}\left(\alpha^T \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}\right) = \dot{\alpha}^T \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} + \alpha^T \begin{bmatrix} \Omega^T & -v \\ 0 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = 0$$

By repeating the above argument we obtain the dynamics for the parameters α as follows

$$\dot{\alpha} = \begin{bmatrix} \Omega^T & 0 \\ v^T & 0 \end{bmatrix} \alpha + c\alpha$$

with c being an arbitrary constant.

Recall the we have defined parameters a as $a_i = \alpha_i/\alpha_4, \quad i = 1,2,3$. Using the above dynamics for α and the definition of a we can write the dynamics for a as follows:

$$\dot{a} = \Omega^T a - (a^T v)a$$

This completes the proof of Proposition 3.2.

If a single camera is used the only available information (measurement) about the curve is the equation of the curve on the image plane given by the first equation in (3.8). Moreover, the parameter vector η can be determined only up to a scale factor. We shall assume that $\eta_6 \neq 0$ which is equivalent to assuming that the projection of the curve does not pass through the origin on the image plane. Therefore, we define new parameters for the curve on the image plane via $\xi_i = \eta_i/\eta_6, \quad i = 1, \ldots, 5$. With these parameters the curve on the image plane is given by

$$\xi_1 x^2 + \xi_2 y^2 + \xi_3 xy + \xi_4 x + \xi_5 y + 1 = 0$$

The dynamics of the parameter vector ξ can be easily derived from the dynamics for η given by (3.9):

$$\begin{aligned} \dot{\xi} &= \begin{bmatrix} \alpha & 0 & -\xi_1\gamma \\ 0 & \beta & -\xi_2\gamma \\ \beta & \alpha & -\xi_3\gamma \\ \gamma & 0 & \alpha - \xi_4\gamma \\ 0 & \gamma & \beta - \xi_5\gamma \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} + \begin{bmatrix} \xi_3\omega_3 - \xi_4\omega_2 - \xi_1\xi_4\omega_2 - \xi_1\xi_5\omega_1 \\ -\xi_3\omega_3 + \xi_5\omega_1 - \xi_2\xi_4\omega_2 - \xi_2\xi_5\omega_1 \\ -2\xi_1\omega_3 + 2\xi_2\omega_3 + \xi_4\omega_1 - \xi_5\omega_2 - \xi_3\xi_4\omega_2 - \xi_3\xi_5\omega_1 \\ 2\xi_1\omega_2 - \xi_3\omega_1 + \xi_5\omega_3 - 2\xi_6\omega_2 - \xi_4^2\omega_2 - \xi_4\xi_5\omega_1 \\ -2\xi_2\omega_1 + \xi_3\omega_2 - \xi_4\omega_3 + 2\xi_6\omega_1 - \xi_5\xi_4\omega_2 - \xi_5^2\omega_1 \end{bmatrix} \\ &\hat{=} W^T(\xi, v)a + \varphi(\xi, \omega) \end{aligned} \tag{3.13}$$

where $\alpha = -2v_1\xi_1 - v_2\xi_2 - v_3\xi_3$, $\beta = -v_1\xi_3 - 2v_2\xi_2 - v_3\xi_4$, $\gamma = -v_1\xi_4 - v_2\xi_5 - 2v_3$.

The parameter vector ξ can be obtained by observing the second order curve on the image plane. Note that the combined system (3.13) and (3.10) is in the form (2.1) for which the Identifier Based Observer is applicable. As before we shall have to verify that the two assumptions are satisfied.

It is reasonable to assume that the boundedness Assumption 2.1 is satisfied because we can assume that the plane containing the curve does not pass too close to the origin in $\mathbb{R}^3$.

We have to check whether the observability Assumption 2.2 is satisfied. This is completely determined by the rank of the matrix $W(\xi, v)W^T(\xi, v)$. Let us now examine when the rank of this matrix drops below 3. Note that the rank of $W(\xi, v)W^T(\xi, v)$ is equal to the rank of $W^T(\xi, v)$ and consider the case $\gamma \neq 0$ The rank of the matrix $W^T(\xi, v)$ is the same as the rank of the matrix obtained from $W^T(\xi, v)$ by multiplying the fourth row by $-\alpha/\gamma$ and adding it to the first row, multiplying the fifth row by $-\beta/\gamma$ and adding it to the second row, and multiplying the fourth row by $-\beta/\gamma$ and fifth row by $-\alpha/\gamma$ and adding both to the third row. The matrix obtained after this manipulation has the rank less then 3 (and the same holds for $W^T(\xi, v)$) if the three equalities below are satisfied simultaneously:

$$\begin{aligned} &\xi_1\gamma^2 - \alpha^2 - \xi_4\alpha\gamma = 0 \\ &\xi_2\gamma^2 - \beta^2 - \xi_5\beta\gamma = 0 \\ &\xi_3\gamma^2 - 2\alpha\beta - \xi_5\alpha\gamma - \xi_4\beta\gamma = 0 \end{aligned}$$

The above equalities are quadratic forms in v and can be rewritten as

$$\begin{aligned} v^T Q_1(\xi) v &= 0 \\ v^T Q_2(\xi) v &= 0 \\ v^T Q_3(\xi) v &= 0 \end{aligned} \tag{3.14}$$

with the matrices Q_i being cubic polynomial matrices in ξ. All three of them are nonsingular for generic values of ξ. From this we conclude that for a generic planar second order curve and for nonzero vector of linear velocities the rank of $W^T(\xi, v)$ is 3.

Remark 3.3: The above procedure can be repeated for third or higher order planar curves. The total number of parameters would increase, but the number of unknown parameters is the same, i.e. 3. Moreover, the structure of the parameter dynamics remains the same.

References

[1] N. Ayache, O.D. Faugeras, "Maintaining Representation of the Environment of a Mobile Robot," Proceedings of the 3rd International Symposium on Robotics Research, 1987, 337-350.

[2] N. Ayache, *Artificial Vision for Mobile Robots*, MIT Press, Cambridge, 1991.

[3] B.K.P. Horn, *Robot Vision*, MIT Press, Cambridge, 1986.

[4] M. Jankovic, B.K. Ghosh, "Visually Guided Ranging from Observations of Points, Lines, and Curves via an Identifier Based Nonlinear Observer," submitted to Systems and Control Letters.

[5] M. Lei, B.K. Ghosh, "New Geometric Methods in Computing the Motion Parameters of a Rigid Body Using Line Correspondences," Proceedings of the American Control Conference, 1992, 1500-1504.

[6] L. Matthies, T. Kanade, R. Szeliski, "Kalman Filter-based Algorithms for Estimating Depth from Image Sequences," International Journal of Computer Vision, 3, 1989, 209-236.

[7] P.K.A. Menon, G.B. Chatterji, B. Sridhar,"A Fast Algorithm for Image-Based Ranging," Proceedings of SPIE International Symposium on Optical Engineering and Photonic in Aerospace Sensing, Orlando, FL, 1991.

[8] K. S. Narendra, A. M. Annaswamy, *Stable Adaptive Systems*, Prentice Hall, Englewood Cliffs, 1989.

[9] B. Sridhar, R. Soursa, B. Hussein,"Passive Range Estimation for Rotorcraft Low Altitude Flight," International Journal of Machine Vision and Applications, 6, 1993, 10-24.

[10] B. Sridhar, R. Soursa, P. Smith, B. Hussein,"Vision-Based Obstacle Detection for Rotorcraft Flight," Journal of Robotic Systems, 9(6), 1992, 709-727.

[11] B. Sridhar, A. Phatak, "Analysis of Image-Based Navigation System for Rotorcraft Low-Altitude Flight," IEEE Transactions on Systems, Man,and Cybernetics, 22, 1992, 290-299.

Signal Design for L-out-of-K CDMA Channels

Abdulaziz A. Alsugair
Roger S. Cheng
Department of Electrical and Computer Engineering
University of Colorado
Boulder, Colorado 80309

Abstract

We consider the symbol synchronous Gaussian L-out-of-K code-division multiaccess channel, and obtain the capacity region and an upper and a lower bound to the symmetric capacity. The lower bound depends on the signature waveforms through the eigenvalues of the product of the crosscorrelation matrix and the diagonal power matrix. We find the sufficient condition for the signature waveform set to maximize this lower bound. An algorithm to construct a set of signature waveforms satisfying this sufficient condition is also given.

I Introduction

In a code-division multiaccess (CDMA) channel, several users simultaneously and independently access a common channel by linearly modulating their discrete-time information symbols with pre-assigned signature waveforms. However, in practical systems, many of these users are idle most of the time and have nothing to send. The L-out-of-K CDMA (LOOK-CDMA) channel models such a situation.

In the LOOK-CDMA channel, there are K independent potential users of which at most L users are simultaneously active. A selector selects at random a subset $S \subseteq \mathcal{K} = \{1, 2, \ldots, K\}$, $|S| \leq L$ of active users. We assume that neither the transmitter nor the receiver knows the set of active users, S. (However, the results are the same if only the receiver knows the set of active users). The k^{th} user modulates its i^{th} information symbol, $X_k(i)$, with a pre-assigned, unit-energy, and time-limited signature waveform, $s_k(t)$ (i.e., $s_k(t) \in L_2[0, T]$ and $\| s_k(t) \| = 1$, $\forall k \in \mathcal{K}$). We consider the symbol synchronous channel where the received signal is the superposition of the modulated signals of the active users embedded in additive white Gaussian noise, $n(t)$, with power spectral density σ^2. When S is the set of active users, the received signal is

$$y_{(S)}(t) = \sum_i \sum_{k \in S} X_k(i) s_k(t - iT) + n(t). \tag{1}$$

This work was partially supported by the National Science Foundation under Grant NCR-92-9812

Each user has an average power constraint which can be written as $E[X_k^2(i)] \leq p_k$, $\forall k \in \mathcal{K}$. Without loss of generality, we assume that $p_1 \leq p_2 \leq \ldots \leq p_K$.

We assume that the set of signature waveforms has dimension $N \leq K$ and is characterized by

$$[s_1(t)\, s_2(t) \ldots s_K(t)]^T = A\underline{\phi}(t), \tag{2}$$

where $A \in \mathbf{C}^{K\times N}$ and $\underline{\phi}(t) \in L_2^N[0,T]$ is an orthonormal basis on the signaling space. Notice that for symbol synchronous channel, if $N \geq K$, the optimal signature waveform set is the orthonormal set and the multiuser channel is decoupled into K single-user channels.

The capacity regions of the symbol synchronous and asynchronous CDMA channels were obtained in [1] and [2], respectively. Optimal signal design for the total and symmetric capacities of the synchronous CDMA channel with equal users' powers was considered in [3]. The capacity region of a random multiaccess channel was found in [4] and that result is used here to obtain the LOOK-CDMA channel capacity.

In Section II, we give the capacity region and the symmetric capacity of the LOOK-CDMA channel. In Sections III and IV, we give an upper and a lower bound, respectively, to the symmetric capacity. The sufficient condition for the signature waveform set to maximize the lower bound is obtained in Section V, where we also give an algorithm to construct a set of signature waveform satisfying the sufficient condition. Finally, we present some numerical examples in Section VI.

II Capacity and Symmetric Capacity

In this section, we obtain the capacity region and the symmetric capacity of the LOOK-CDMA channel.

Theorem 1

The capacity region of the LOOK-CDMA channel (1) is equal to

$$C = \left\{(R_1,\ldots,R_K) : \sum_{k\in S} R_k \leq \frac{1}{2}\log[\det(I_{|S|} + \hat{H}_S)],\ \forall S \subseteq \mathcal{K}, |S| \leq L\right\}, \tag{3}$$

where $\hat{H} = W^{\frac{1}{2}} H W^{\frac{1}{2}}$, $W = \text{diag}\,[w_1 w_2 \ldots w_K]$, $w_k = p_k/\sigma^2$, and H is the crosscorrelation matrix of the signature waveforms, i.e.

$$H_{lk} = \int_0^T s_l(t) s_k^*(t)dt \quad \forall l,k \in \mathcal{K}, \tag{4}$$

or, equivalently, $H = AA^*$. For any matrix M, the notation M_S is the $|S|\times|S|$ submatrix of M formed by the i^{th} row and the j^{th} column of M for all $i,j \in S$. □

The proof is a straightforward application of the result in [4] and is omitted here.

The symmetric capacity is defined as

$$C_{sym} = \max_{(R,\ldots,R)\in C} R, \tag{5}$$

and it gives the maximum rate at which all active users in S can transmit reliably, $\forall\, S \subseteq \mathcal{K}$ and $|S| \leq L$.

Using Theorem 1, we can write C_{sym} as

$$\begin{aligned} C_{sym} &= \min_{S\subseteq\mathcal{K},|S|\leq L} \frac{1}{2|S|}\log[\det(I_{|S|}+\hat{H}_S)] && (6) \\ &= \min_{S\subseteq\mathcal{K},|S|\leq L} \frac{1}{2|S|}\sum_{i=1}^{|S|}\log[1+\lambda_i^{\{S\}}], && (7) \end{aligned}$$

where $0 \leq \lambda_1^{\{S\}} \leq \ldots \leq \lambda_{|S|}^{\{S\}}$ are the eigenvalues of $\hat{H}_S$ which has diagonal entries $\underline{w}^{\{S\}} = [w_1^{\{S\}} \ldots w_{|S|}^{\{S\}}]$ arranged in nondecreasing order. Also, using the fact that the diagonal entries of a Hermitian matrix majorizes its eigenvalues [5, p.193], we have

$$\sum_{i=1}^{n}\lambda_i^{\{S\}} \leq \sum_{i=1}^{n} w_i^{\{S\}}, \tag{8}$$

for $n = 1, \ldots, |S|$ with equality when $n = |S|$.

III Upper Bound to the Symmetric Capacity

In this section, we obtain an upper bound to the symmetric capacity of the LOOK-CDMA channel. In order to obtain the upper bound, we need the following two lemmas.

Lemma 1

For any nondecreasing $\underline{\lambda}, \underline{\tilde{\lambda}} \in \mathbb{R}_+^l$ (i.e. $0 \leq \lambda_1 \leq \ldots \leq \lambda_l$ and $0 \leq \tilde{\lambda}_1 \leq \ldots \leq \tilde{\lambda}_l$) such that $\sum_{i=1}^{l}\lambda_i = \sum_{i=1}^{l}\tilde{\lambda}_i$, if

$$\tilde{\lambda}_i \geq \lambda_i \qquad i = 1, \ldots, j, \tag{9}$$

and

$$\tilde{\lambda}_i \leq \lambda_i \qquad i = j+1, \ldots, l, \tag{10}$$

for some $0 < j < l$, then

$$\sum_{i=1}^{l}\log(1+\tilde{\lambda}_i) \geq \sum_{i=1}^{l}\log(1+\lambda_i). \tag{11}$$

□

Lemma 2

For any nondecreasing $\underline{v} \in \mathbb{R}_+^l$ and $0 \leq r < l$, let

$$\underline{\tilde{\lambda}}(\underline{v}, r) = \arg \max_{\substack{\underline{\lambda} = [0\ldots0\lambda_{r+1}\ldots\lambda_l] \\ \underline{v} \text{ majorizes } \underline{\lambda}}} \sum_{i=r+1}^{l}\log[1+\lambda_i]. \tag{12}$$

Then, we have

$$\begin{aligned} \tilde{\lambda}_{r+1}(\underline{v}, r) &= \min_{n\in\{r+1,\ldots,l\}}\left\{\frac{1}{n-r}\sum_{i=1}^{n} v_i\right\}, && (13) \\ \tilde{\lambda}_{r+1}(\underline{v}, r) = \tilde{\lambda}_{r+2}(\underline{v}, r) &= \ldots = \tilde{\lambda}_{\tilde{n}}(\underline{v}, r), && (14) \\ \tilde{\lambda}_k(\underline{v}, r) &= v_k, \forall k > \tilde{n}, && (15) \end{aligned}$$

where $\tilde{n} = \arg\ \min_{n\in\{r+1,\ldots,l\}}\left\{\frac{1}{n-r}\sum_{i=1}^{n} v_i\right\}$. If $r = 0$, then $\underline{\tilde{\lambda}}(\underline{v}, r) = \underline{v}$. □

Theorem 2

The symmetric capacity is upper bounded by

$$\bar{C}_{sym} = \begin{cases} \frac{1}{2}\log(1+w_1) & L \leq N, \\ \min_{l=1,\dots,L} \frac{1}{2l}\sum_{i=1}^{l} \log[1+\tilde{\lambda}_i(\underline{w}^{\{\mathcal{L}\}}, r)] & L > N, \end{cases} \tag{16}$$

where $r = \max\{0, (l-N)\}$, and $\mathcal{L} = \{1, 2, \dots, l\}$. □

Proof: We can rewrite (7) as

$$C_{sym} = \min_{l=1,2,\dots,L} \min_{S \subseteq \mathcal{K}, |S|=l} \frac{1}{2l}\sum_{i=1}^{l} \log[1+\lambda_i^{\{S\}}]. \tag{17}$$

Since $\hat{H}$ has rank N, $\hat{H}_S$ has rank at most N and has at most N nonzero eigenvalues. Then, from Lemma 2 and the fact that $\underline{w}^{\{S\}}$ majorizes $\underline{\lambda}^{\{S\}}$, we have

$$C_{sym} \leq \min_{l=1,2,\dots,L} \min_{S \subseteq \mathcal{K}, |S|=l} \frac{1}{2l}\sum_{i=1}^{l} \log[1+\tilde{\lambda}_i(\underline{w}^{\{S\}}, r)], \tag{18}$$

where $r = \max\{0, (l-N)\}$. Since $\sum_{i=1}^{l} w_i^{\{S\}} \geq \sum_{i=1}^{l} w_i^{\{\mathcal{L}\}} = \sum_{i=1}^{l} w_i, \forall\, l = 1, \dots L$, we have

$$C_{sym} \leq \min_{L=1,\dots,L} \frac{1}{2l}\sum_{i=1}^{l} \log[1+\tilde{\lambda}_i(\underline{w}^{\{\mathcal{L}\}}, r)]. \tag{19}$$

When $L \leq N$, $r = 0$ and we have

$$\begin{aligned} C_{sym} &\leq \min_{l=1,2,\dots,L} \frac{1}{2l}\sum_{i=1}^{l} \log[1+w_i], && (20) \\ &= \log[1+w_1]. && (21) \end{aligned}$$

∎

Unlike the C_{sym} which depends on the signature waveform, the upper bound in Theorem 2 depends only on the signal-to-noise ratios and the dimension of the signaling space. Moreover, it is independent of K.

When $L \leq N$, the bound is trivial since the symmetric capacity is always less than the single-user capacities of all the users. When $L > N$, the bound is not trivial since it is strictly less than all the single-user capacities.

The following Corollary gives an upper bound for the equal power case.

Corollary 1

If $w_1 = w_2 = \dots = w_K = w$ then

$$\bar{C}_{sym} = \begin{cases} \frac{1}{2}\log(1+w) & L \leq N \\ \frac{N}{2L}\log(1+\frac{L}{N}w) & L > N \end{cases} \tag{22}$$

□

IV Lower Bound to the Symmetric Capacity

In this section, we obtain a lower bound to the symmetric capacity of the LOOK-CDMA channel. Before we state the main result of this section, we need the following lemma.

Lemma 3

For any nondecreasing $\underline{\lambda} \in \mathbb{R}_+^K$, let

$$\underline{\alpha}^*(\underline{\lambda}, l, d) = \arg\min_{\underline{\alpha}} \sum_{i=1}^{l} \log[1 + \alpha_i], \tag{23}$$

where the minimization is taken over all nondecreasing $\underline{\alpha} \in \mathbb{R}_+^l$ such that

$$\sum_{i=1}^{l} \alpha_i = d, \tag{24}$$

and

$$\lambda_i \leq \alpha_i \leq \lambda_{K-l+i}. \tag{25}$$

Then there exists an integer M, $0 \leq M \leq l$, such that

$$\alpha_i^*(\underline{\lambda}, l, d) = \begin{cases} \lambda_i & \text{if } i = 1, \ldots, M, \\ \lambda_{K-l+i} & \text{if } i = M+1, \ldots, l, \\ \alpha_M^*(\underline{\lambda}, l, d) & \text{if } i = M, \end{cases} \tag{26}$$

where $\alpha_M^*(\underline{\lambda}, l, d)$ is chosen to satisfy (24). □

Theorem 3

The symmetric capacity is lower bounded by

$$\underline{C}_{sym}(\underline{\lambda}, \underline{w}) = \min_{l=1,2,\ldots,L} \frac{1}{2l} f(\underline{\lambda}, l, \underline{w}), \tag{27}$$

where

$$f(\underline{\lambda}, l, \underline{w}) = \sum_{i=1}^{l} \log[1 + \alpha_i^*(\underline{\lambda}, l, \sum_{i=1}^{l} w_i)], \tag{28}$$

and $\underline{\alpha}^*$ is defined in (26). In (27), $\underline{\lambda}$ are the eigenvalues of $\hat{H}$. □

Proof: Using the inclusion principle on the eigenvalues of a Hermitian matrix and its submatrix [5, p.189], we have that the eigenvalues of $\hat{H}_S$ must satisfy

$$\lambda_i \leq \lambda_i^{\{S\}} \leq \lambda_{K-|S|+i}, \tag{29}$$

where $\lambda_1 \leq \ldots \leq \lambda_K$ are the eigenvalues of $\hat{H}$. Hence, using Lemma 3 and noting that $d^{\{S\}} = \sum_{i=1}^{l} w_i^{\{S\}}$, we have, from (17) and (29),

$$C_{sym} \geq \min_{l=1,2,\ldots,L} \min_{S \subseteq \mathcal{K}, |S|=l} \frac{1}{2l} \sum_{i=1}^{l} \log[1 + \alpha_i^*(\underline{\lambda}, l, d^{\{S\}})], \tag{30}$$

$$= \min_{l=1,2,\ldots,L} \frac{1}{2l} \sum_{i=1}^{l} \log[1 + \alpha_i^*(\underline{\lambda}, l, \sum_{i=1}^{l} w_i)] \tag{31}$$

where the last equation follows from the fact that $\sum_{i=1}^{l} \log[1 + \alpha_i^*(\underline{\lambda}, l, d)]$ is decreasing in d. ∎

Note that the symmetric capacity C_{sym} depends on the matrix $\hat{H}$ in a complicated manner. In order to evaluate C_{sym}, one has to calculate the log [det($\cdot$)] over $\sum_{l=1}^{L} \binom{K}{l}$ subsets of $\mathcal{K}$. Theorem 3 gives a lower bound to the symmetric capacity that depends on the signature waveforms through the eigenvalues of $\hat{H}$. As we will see in the next section, this fact allows us to design signature waveform set that maximizes the lower bound.

V Signature Waveform Design

We now restrict our attention to the design of signature waveform set that maximizes the lower bound, $\underline{C}_{sym}$, which is a function of the eigenvalues of $\hat{H}$ and the signal-to-noise ratios of the users. Note that $\underline{\lambda}$ is the vector of eigenvalues of $\hat{H}$ whose diagonal entries are $\underline{w}$ and hence $\underline{w}$ majorizes $\underline{\lambda}$. We will show in Theorem 5 that for any $\underline{\lambda}$ and $\underline{w}$ such that $\underline{w}$ majorizes $\underline{\lambda}$, there exists a matrix A such that $W^{\frac{1}{2}}AA^*W^{\frac{1}{2}}$ has diagonal entries $\underline{w}$ and eigenvalues $\underline{\lambda}$. Hence, the problem of maximizing $\underline{C}_{sym}(\underline{\lambda}, \underline{w})$ over the signature waveform set is equivalent to that of maximizing $\underline{C}_{sym}(\underline{\lambda}, \underline{w})$ over all $\underline{\lambda}$ majorized by $\underline{w}$.

We will first find the set of eigenvalues that maximizes the bound $\underline{C}_{sym}(\underline{\lambda}, \underline{w})$ and then exhibit an algorithm to find the signature waveform set that achieves the bound.

Lemma 4

For any nondecreasing $\underline{\lambda}$, $\tilde{\underline{\lambda}}$ and $\underline{w} \in \mathbb{R}_+^K$ such that $\underline{w}$ majorizes both $\underline{\lambda}$ and $\tilde{\underline{\lambda}}$, if

$$\tilde{\lambda}_i \geq \lambda_i \qquad i = 1, \ldots, J, \tag{32}$$

and

$$\tilde{\lambda}_i \leq \lambda_i \qquad i = J+1, \ldots, K, \tag{33}$$

for some $0 < J < K$, then

$$f(\tilde{\underline{\lambda}}, l, \underline{w}) \geq f(\underline{\lambda}, l, \underline{w}) \qquad \forall l = 1, \ldots, K, \tag{34}$$

where $f(\underline{\lambda}, l, \underline{w})$ is defined in (28). □

Theorem 4

The sufficient condition for the signature waveform set to maximize the lower bound $\underline{C}_{sym}(\underline{\lambda}, \underline{w})$ is that $\hat{H} = W^{\frac{1}{2}}HW^{\frac{1}{2}}$ has eigenvalues $\tilde{\underline{\lambda}}$, where $\tilde{\underline{\lambda}}$ is defined by

$$\tilde{\lambda}_{K-N+1} = \min_{n \in \{K-N+1, \ldots, K\}} \Big\{\frac{1}{n-K+N} \sum_{i=1}^{n} w_i\Big\}, \tag{35}$$

$$\tilde{\lambda}_{K-N+1} = \tilde{\lambda}_{K-N} = \ldots = \tilde{\lambda}_{\tilde{n}}, \tag{36}$$

and

$$\tilde{\lambda}_k = w_k, \qquad k = \tilde{n}+1, \ldots, K, \tag{37}$$

where

$$\tilde{n} = \arg \min_{n \in \{K-N+1, \ldots, K\}} \Big\{\frac{1}{n-K+N} \sum_{i=1}^{n} w_i\Big\}. \tag{38}$$

□

Proof: From Lemmas 2 and 4, it is straightforward to show that

$$\tilde{\lambda} = \arg\sup_{\underline{\lambda}} f(\underline{\lambda}, l, \underline{w}). \tag{39}$$

Since $\underline{\tilde{\lambda}}$ is independent of l, then $\underline{\tilde{\lambda}}$ maximizes $\underline{C}_{sym}(\underline{\lambda}, \underline{w})$. $\blacksquare$

Corollary 2

If $w_1 = w_2 = \ldots = w_K = w$, then

$$\tilde{\lambda}_i = \begin{cases} 0 & i = 1, \ldots, K-N, \\ w\frac{K}{N} & i = K-N+1, \ldots, K, \end{cases} \tag{40}$$

and

$$\underline{C}_{sym}(\underline{\tilde{\lambda}}, \underline{w}) = \frac{N}{2K}\log[1 + w\frac{K}{N}] + \min_{l=1,\ldots,L}\frac{1}{2l}\Big[\log[1 + \beta w\frac{K}{N}] - \beta\log[1 + w\frac{K}{N}]\Big], \tag{41}$$

where

$$\beta = \frac{lN}{K} - \left\lfloor \frac{lN}{K} \right\rfloor. \tag{42}$$

The second term in the right hand side of (41) is nonnegative and is equal to 0 iff $\beta = 0$. If we write $\frac{K}{N} = \frac{p}{q}$, where p and q are co-prime, the second term in (41) is equal to 0 iff $L \geq p$. $\square$

In the equal power case in Corollary 2, the sufficient condition coincides with that obtained in [3]. Hence, the signature waveform set that maximizes $\underline{C}_{sym}(\underline{\lambda}, \underline{w})$ for the LOOK-CDMA channel also maximizes the symmetric capacity, as well as the total capacity, of the CDMA channel. The maximum symmetric capacity of the CDMA channel obtained in [3] is equal to $\frac{N}{2K}\log[1 + w\frac{K}{N}]$. Compairing to the lower bound in (41), we find that the symmetric capacity of the LOOK-CDMA channel is strictly larger than that of the CDMA channel if $L < p$, where p is defined in Corollary 2.

We now want to construct a signature waveform matrix A such that $W^{\frac{1}{2}}AA^{*}W^{\frac{1}{2}}$ has diagonal entries $\underline{w}$ and eigevalues $\underline{\lambda}$. The construction is not unique and we exhibit an algorithm to do this in at most $(K-1)$ steps.

Let us define, for each $i < j$ and $\alpha \in [0, 1]$ a unitary matrix

$$J_{mn}(i, j, \alpha) = \begin{cases} \sqrt{\alpha} & i, j = m = n, \\ \sqrt{1-\alpha} & i = m, j = n, \\ -\sqrt{1-\alpha} & j = m, i = n, \\ 1 & i, j \neq m = n, \\ 0 & \text{otherwise.} \end{cases} \tag{43}$$

In other words, $J(i, j, \alpha)$ is equal to the identity matrix I except that $J_{ii}(i, j, \alpha) = J_{jj}(i, j, \alpha) = \sqrt{\alpha}$ and $J_{ij}(i, j, \alpha) = -J_{ji}(i, j, \alpha) = \sqrt{1-\alpha}$.

Theorem 5

For any nondecreasing $\underline{\lambda} = [0 \ldots 0\lambda_{K-N+1} \ldots \lambda_K]$ and $\underline{w} \in \mathbb{R}_+^K$ such that $\underline{w}$ majorizes $\underline{\lambda}$, there exists $A \in \mathbb{R}^{K \times N}$ such that $\hat{H} = W^{\frac{1}{2}}AA^TW^{\frac{1}{2}}$ has diagonal entries $\underline{w}$ and eigenvalues $\underline{\lambda}$. $\square$

Proof: We will prove the theorem by constructing an algorithm that gives the desired matrix.

Algorithm

Step 1: i=1, $E^{(0)} = \text{ diag } [\lambda_1 \ldots \lambda_K]$.

Step 2:

$$l_i = \arg \min_{\substack{n > i \\ E^{i-1}_{nn} \geq w_i}} E^{i-1}_{nn},$$

$$\alpha_i = \frac{w_i - E^{(i-1)}_{l_i l_i}}{E^{(i-1)}_{ii} - E^{(i-1)}_{l_i l_i}},$$

$$\tilde{E}^{(i)} = J(i, l_i, \alpha_i)\, E^{(i-1)}\, J^T(i, l_i, \alpha_i).$$

Step 3: Find j such that $i < j \leq l_i$ and $\tilde{E}^{(i)}_{j-1,j-1} \leq \tilde{E}^{(i)}_{l_i l_i} \leq \tilde{E}^{(i)}_{jj}$,

$E^{(i)} = P^{(i)}\, \tilde{E}^{(i)}\, P^{(i)T}$, where

$$P^{(i)}_{mn} = \begin{cases} 1 & \text{if } m = n < j \text{ and } m = n \geq l_i, \\ 1 & \text{if } l_i - 1 \geq m - 1 = n \geq j, \\ 1 & \text{if } m = j, n = l_i, \\ 0 & \text{otherwise.} \end{cases} \tag{44}$$

Step 4: If $i = K - 1$ go to step 5,
otherwise, $i \leftarrow i + 1$, go to step 2

Step 5:

$$U = P^{(K-1)} J(K-1, l_{K-1}, \alpha_{K-1}) \ldots P^{(2)} J(2, l_2, \alpha_2) P^{(1)} J(1, l_1, \alpha_1),$$

$$A = W^{-\frac{1}{2}} U \breve{\Lambda},$$

and

$$\hat{H} = E^{(K-1)} = W^{\frac{1}{2}} A A^T W^{\frac{1}{2}},$$

where

$$\breve{\Lambda}_{ij} = \begin{cases} \sqrt{\lambda_i} & i - j = K - N, \\ 0 & \text{otherwise.} \end{cases}$$

Stop.

After the i^{th} iteration of the algorithm, the matrix $E^{(i)}$ has diagonal elements ordered in nondecreasing order. Moreover, $w_1 \ldots w_i$ are the first i diagonal elements. The submatrix which consists of the $i + 1^{st}$ to the K^{th} rows and columns is diagonal, and $\sum_{j=1}^{n} E^{(i)}_{jj} \leq \sum_{j=1}^{n} w_j$, for $n = 1, \ldots, K$. Therefore, α_{i+1} is guaranteed to be in the range $[0, 1]$ and hence $\tilde{E}^{(i+1)}_{i+1,i+1} = w_{i+1}$. Since $J(i, l_i, \alpha_i)$ and $P^{(i)}$ are unitary matrices $\forall i = 1, \ldots, K-1$, U is also unitary. Finally, $\hat{H} = E^{(K-1)} = U E^{(0)} U^T$ has diagonal entries $\underline{w}$ and eigenvalues $\underline{\lambda}$. ∎

VI Numerical Results

In this section, we present some numerical examples to illustrate the bounds to the symmetric capacity.

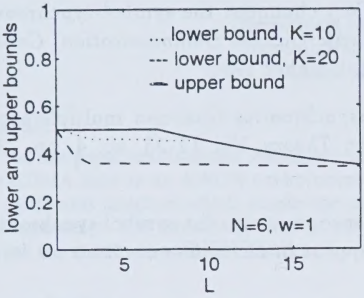


Figure 1: $\bar{C}_{sym}$ and $\underline{C}_{sym}$ versus L for the equal users powers

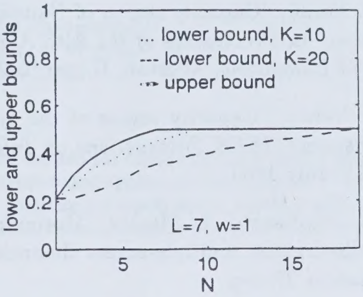


Figure 2: $\bar{C}_{sym}$ and $\underline{C}_{sym}$ versus N for the equal users powers

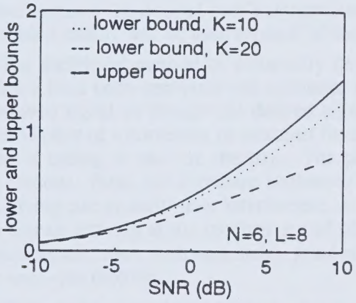


Figure 3: $\bar{C}_{sym}$ and $\underline{C}_{sym}$ versus the SNR for the equal users powers

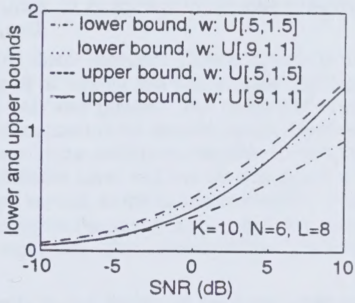


Figure 4: $\bar{C}_{sym}$ and $\underline{C}_{sym}$ versus the average SNR for the unequal users powers

In the first three examples, we consider the equal power case with $K = 10$ and $K = 20$. Note that $\bar{C}_{sym}$ is independent of K. In Figure 1, we plot $\bar{C}_{sym}$ and the maximum $\underline{C}_{sym}$, maximized over all signature waveform set, versus L, the maximum number of simultaneous active users, for $N = 6$, the signal space dimension. We note that the gap between the two bounds is smaller for $K = 10$ and they coincide when $L = K$. In Figure 2, we plot the bounds versus N for $L = 7$. As N gets larger, the gap between the bounds becomes smaller until they coincide when $N = K$, in which case orthogonal signaling is possible. In Figure 3, we plot the bounds versus the signal-to-noise ratio (SNR) for $N = 6$ and $L = 8$. For small SNR, the two bounds are very close and the gap between the bounds increases with the SNR.

The last example illustrates the unequal power case. In Figure 4 , we plot $\bar{C}_{sym}$ and $\underline{C}_{sym}$ versus the average SNR for $K = 10, N = 6$, and $L = 8$. The users' powers, $\underline{p}$ are randomly drawn from a uniform distribution of mean 1. Two distributions are considered: $U[.5, 1.5]$ and $U[.9, 1.1]$. Note that the gap between the bounds is smaller as the variance in the power distribution decreases.

References

[1] S. Verdú, "Capacity region of Gaussian CDMA channels: the symbol-synchronous case," in *Proceedings of the 24th Allerton Conference on Communication, Control and Computing*, Allerton, IL, pp. 1025–1034, October 1986.

[2] S. Verdú, "Capacity region of the symbol-asynchronous Gaussian multiple-access channel," *IEEE Transactions on Information Theory*, Vol. IT-35, no. 4, pp. 735–751, July 1989.

[3] M. Rupf and J. L. Massey, "Optimum sequence multisets for symbol-synchronous code-division multiple-access channels," to appear in *IEEE Transactions on Information Theory.*

[4] A. Satt and E. Plotnik, "Decoding rule and error exponent bound for the random multiple-access channel," preprint, July 1992.

[5] R. A. Horn and C. R. Johnson, *Matrix Analysis.* Cambridge University Press, Cambridge, 1990.

Trellis-Based Multiuser Receivers for Convolutionally Coded CDMA Systems

T.R. Giallorenzi
S.G. Wilson

Department of Electrical Engineering
Thornton Hall, University of Virginia
Charlottesville, VA 22903

Abstract

In this paper, trellis-based multiuser receivers are studied for convolutionally coded asynchronous CDMA links in an AWGN environment. Two trellis-based approaches are proposed. The first is a partitioned structure which attacks the multiuser interference equalization and decoding operations separately. The second approach is the maximum likelihood sequence estimator for the problem and is a combined equalizer and decoder.

1. Introduction

In code division multiple access (CDMA) systems, multiple users transmit, often asynchronously, over a common communication channel, typically using the direct sequence spread spectrum technique. The receiver operating in the AWGN environment receives a signal which is the sum of all of the transmitted signals in noise, and the receiver must synchronize to the desired signal or signals and estimate the desired user's transmitted data. Often, in an attempt to improve performance, error control coding will be used on each of the links as well.

The traditional method of coherently demodulating direct sequence CDMA signals is to synchronize a local code generator and oscillator to the signal of interest and then to make decisions on the received signal as though the desired signal is the only one present. The traditional decoder's structure is that of a correlator or matched filter which is matched to the desired signal, followed by a decoder if coding is used on the link. The performance of the traditional decoder suffers for two major reasons. First, the signature sequences of the different users will not be orthogonal to each other, giving rise to multi-user interference, or MUI, and second, in the common situation where all of the signals arriving at the receiver are of different strengths the strong signals tend to overwhelm the weak signals, even with reasonably good signature sequences. This second problem is referred to as the *near–far* problem.

There has been a large amount of interest recently in the design of multiuser receivers for CDMA systems. These receivers jointly estimate the transmitted symbols of all of the users in the system, as opposed to estimating them independently. This approach is most appropriate for a base station in a multipoint-to-point network where the receiver must acquire and demodulate all of the signals in the network. Almost all of the multiuser detection work has centered on uncoded links, see [1] - [5]. Only recently has the problem of multiuser detection of coded links been considered, [6] and [11]. In [6], a linear equalization technique was considered and the authors referred to the use of a convolutional code as well. In [11], a number of multistage decision feedback approaches were examined along with some variations of the trellis-based approaches that will be discussed in this paper.

In this paper, trellis-based MUI equalization and decoding techniques will be examined for convolutionally coded links. Section 2 of this paper will outline some of the notation that will be used, and will also define the problem in more precise terms. In section 3 trellis-based equalization and decoding techniques will be discussed. Two decoders will be proposed, a partitioned trellis-based receiver, which treats the equalization and decoding operations separately, and an optimum sequence estimator, which is the maximum likelihood sequence estimation receiver for this problem.

2. Notation

It will be assumed that the CDMA system has K users operating in an asynchronous fashion. In general, it will be assumed that each user employs convolutional coding on its link. While it is quite conceivable that block codes could be used effectively on a CDMA link, convolutional codes have the advantage that they operate in a sequential fashion. Because the decoders that will be studied in this work are sequential in nature, the convolutional codes are a much better match to the decoders than block codes.

At each time interval, the convolutional code is generated for user k by passing P binary information bits, $\overline{I}_k(n) = (I_k^{(1)}(n),\ldots, I_k^{(P)}(n))$, through a shift register consisting of W stages with Q modulo-2 adders. The number of output bits for each P-bit input sequence is Q bits. The rate of the code is $R_c = P/Q$ and the constraint length of the code is W. The output sequence of binary code bits for the interval corresponding to input bits $\overline{I}_k(n)$ is $(D_k^{(1)}(n),\ldots,D_k^{(Q)}(n))$. Note that for $W = 1$ and $P = Q = 1$, we have the uncoded case, so in that case $D_k(n) = I_k(n)$.

In the time interval $[nT_b+\theta_k,(n+1)T_b+\theta_k)$, user k transmits data bit $D_k^{(1)}(n)$, where θ_k represents the time shift of the k^{th} user relative to some reference time, thus accounting for the asynchronism of the users relative to each other. T represents the code bit period and $T_b = T/R_c$ is the information bit duration. Let $\theta_k = m_k T+\tau_k$, $\tau_k \in [0,T)$, and $m_k \in \{0,\ldots,Q-1\}$. Thus $m_k T$ is a coarse time shift and τ_k is a fine time shift for user k.

Each user in the system is assigned a particular signature sequence, and it will be assumed that this signature sequence has a duration equal to the code bit interval, although this assumption can be relaxed with a change of the notation. We will combine the carrier and signature sequence into a single signal, thus the k^{th} carrier multiplied by the binary signature sequence, $PN_k(t)$, will be denoted by

$$s_k(t-nT-\tau_k) = \sqrt{2/T}\; PN_k(t-nT-\tau_k)\, cos\,(\omega_c(t-nT-\tau_k)) \qquad nT+\tau_k \le t \le (n+1)T+\tau_k \tag{1}$$

and it will have a time support limited to the time interval $[nT+\tau_k,(n+1)T+\tau_k)$. The energy of the k^{th} user's code bit measured at the receiver will be denoted by E_k. It will be assumed that all K users transmit their signals through a common additive white Gaussian noise channel, and so the received signal will have the following form:

$$r(t)= \sum_{i=-\infty}^{\infty} \sum_{k=1}^{K} \sum_{j=1}^{Q} D_k^{(j)}(i)\sqrt{E_k}\; s_k(t-(i+j-1)T-\theta_k) + n(t) \ . \tag{2}$$

Next we define the partial cross-correlation of signature sequences j and k to be:

$$\rho_{jk}(i) = \int_{-\infty}^{\infty} s_j(t-\tau_j)\, s_k(t-iT-\tau_k)\, dt \ . \tag{3}$$

It is worth noting that $\rho_{jj}(0)=1$ and $\rho_{jk}(i)=\rho_{kj}(-i)$.

The base station that will be referred to as the *conventional* base station on a coded link attempts to estimate the j^{th} user's data operating with only the matched filter outputs for the j^{th} user. It will be assumed that the Viterbi algorithm operating on each channel's observed code symbols is a soft-Viterbi algorithm operating with a decoding delay of δ information symbols, where generally δ will be several times the constraint length, W, of the code. The time complexity per decoded bit for this receiver may be estimated by considering the number of metric computations per information bit decided. If we define $\kappa = \log_2 S$ where S is the number of states of each user's encoder, then there are $2^{\kappa+P}$ metrics computed for every P bits decided, so $TCB = O(2^{\kappa+P}/P)$.

As in the uncoded case, there are a number of ways that a multiuser receiver can operate to improve upon the performance of a conventional basestation. In [6] a sliding window version of the decorrelating detector, which was the linear decoder first proposed in [3], was studied and the authors alluded to the use of a convolutional code on the link. In that approach, a linear detector, whose decision statistic is a linear combination of the matched filter outputs, is used to equalize the MUI prior to decoding with a bank of K Viterbi decoders. In [11], a number of multistage decision feedback approaches are examined. In the next section, both a partitioned trellis-based approach and the optimum maximum likelihood sequence estimator will be derived for this problem.

3. Trellis-Based Receivers

The standard decoder for a convolutional code is the Viterbi algorithm operating on the encoder's trellis because it allows maximum likelihood sequence estimation with a modest decoder complexity when the constraint length of the code is small. The optimal joint decoder for an uncoded asynchronous CDMA system is also a Viterbi algorithm operating on a time-varying trellis with 2^{K-1} states, [1]. The simplest trellis-based approach is therefore the decoder in which an optimal equalizer for the MUI operates at the code symbol level and feeds the resulting hard code symbol estimates to a bank of Viterbi algorithms which operate independently of each other. (see Figure 1a) This will be referred to as the *partitioned* trellis-based solution.

The time complexity per decoded bit for this receiver may be estimated by again considering the number of metric computations per information bit decided. The optimal equalizer computes 2^K metrics for every code bit that is decided and there are again 2^{K+P} metrics computed in the decoding Viterbi algorithm for every P bits decided, so the total TCB = $O([Q2^K + 2^{K+P}]/P)$. (Note that this assumes that the number of multiplications and additions is roughly the same in the equalizer's metric and the decoder's metric.)

Another variation of the partitioned trellis-based decoder would be to modify the equalizer Viterbi algorithm to supply soft-decision outputs to the outer decoding Viterbi algorithms. This could be accomplished using the techniques described in [10]. It is reasonable to expect that the performance of this receiver would be slightly better than the standard partitioned trellis-based receiver in most cases.

One major limitation of the partitioned trellis-based decoder approach, of the soft or hard-decision variety, is that the equalization of the MUI will have to take place at the code symbol level without exploiting the coding. It is reasonable to expect that the decoder which takes the coding into account and effectively operates at the information symbol level can attain a higher performance level.

3.1 Optimum Sequence Estimator For Rate-1/2 Convolutional Codes

The optimal MLSE will now be derived for the special case in which each user in the network is employing a rate-1/2 convolutional code with a constraint length of W. Our limitation to this special case will facilitate the derivation of the decoder considerably, and it will then be outlined how the optimal decoder can be derived in a similar way for a general rate-P/Q convolutional code case.

To begin, it is important to note that the optimal sequence estimator or equalizer for multiple-user uncoded signals operates in a "round-robin" fashion among all K users in the system, [1]. This Viterbi algorithm traverses one trellis stage per channel bit observed. The optimal sequence estimator for decoding the rate-1/2 code for one of the users in a single user environment, however, is a Viterbi algorithm which requires two channel observations from the user of interest to move ahead one stage in the trellis. The fact that the equalizer and decoder operate in a fundamentally different way suggests that a slightly different view of the problem is required. The following view of the problem will be adopted in order to bypass this issue. The rate-1/2 convolutional code can be viewed not as a code which produces two binary bits per information bit period, T_b, but as an equivalent trellis code which produces one 4-ary coded waveform every T_b. By formulating the equalization problem at the receiver with respect to this super-code-symbol view of the received signal, we can accomplish both the tasks of equalization and decoding in the same Viterbi algorithm. Because there is only one 4-ary super-symbol received for each information bit that must be decided, the decoder can be formulated in basically the same fashion as was used in [1] for the MUI problem or [7] for the ISI problem.

We begin by defining the following notation.

$$g_1(t) = \begin{bmatrix} 1 & t \,\varepsilon\, [0,T) \\ 0 & \mathit{otherwise} \end{bmatrix} \tag{4}$$

$$g_2(t) = \begin{bmatrix} 1 & t \,\varepsilon\, [T,2T) \\ 0 & \mathit{otherwise} \end{bmatrix} \tag{5}$$

Next, the following function may be defined

$$\tilde{D}_k(t-iT_b-\theta_k) = D_k^{(1)}(i)\, g_1(t-iT_b-\theta_k) + D_k^{(2)}(i)\, g_2(t-iT_b-\theta_k) \tag{6}$$

Two of the signature-carrier waveforms can be concatenated to form a super-signature waveform.

$$\tilde{s}_k(t-iT_b-\theta_k) = s_k(t-iT_b-\theta_k) + s_k(t-(i+\tfrac{1}{2})T_b-\theta_k) \tag{7}$$

The received waveform may now be written in terms of these waveforms of duration T_b:

$$r(t) = \sum_{i=-\infty}^{\infty}\sum_{k=1}^{K} \tilde{D}_k(t-iT_b-\theta_k)\,\tilde{s}_k(t-iT_b-\theta_k)\sqrt{E_k} + n(t) \tag{8}$$

This signal may be viewed as a four-valued super-code symbol, $\{D_k^{(1)}(i), D_k^{(2)}(i)\}$, modulating a pair of orthonormal basis functions through the procedure defined above. The basis functions in this new view of the waveform are

$$\phi_{1k}(t) = g_1(t)\,\tilde{s}_k(t) \tag{9}$$

and

$$\phi_{2k}(t) = g_2(t)\,\tilde{s}_k(t) \tag{10}$$

appropriately synchronized with the information bit periods. Thus, this equivalent view of the coding process suggests that the information bits are mapped by the encoder onto waveforms in a space defined by $\phi_{1k}(t)$ and $\phi_{2k}(t)$. Note that although the bases defined in (9) and (10) are orthonormal, they are not, in general, orthogonal to $\phi_{1j}(t)$ and $\phi_{2j}(t)$ which are the basis set for another user in the system, user j, since $s_k(t)$ and $s_j(t)$ are not orthogonal in general. The result when the received signal is a sum of K component signals is MUI. We now define four parameters which are a measure of the degree of correlation between the basis functions of the different users.

$$\tilde{\rho}_{jk}(l) = \int_{-\infty}^{\infty} \phi_{1j}(t-\theta_j)\,\phi_{1k}(t-lT_b-\theta_k)\,dt \tag{11}$$

$$\tilde{U}_{jk}(l) = \int_{-\infty}^{\infty} \phi_{2j}(t-\theta_j)\,\phi_{2k}(t-lT_b-\theta_k)\,dt \tag{12}$$

$$\tilde{V}_{jk}(l) = \int_{-\infty}^{\infty} \phi_{1j}(t-\theta_j)\,\phi_{2k}(t-lT_b-\theta_k)\,dt \tag{13}$$

$$\tilde{W}_{jk}(l) = \int_{-\infty}^{\infty} \phi_{2j}(t-\theta_j)\,\phi_{1k}(t-lT_b-\theta_k)\,dt \tag{14}$$

These parameters play the same role in the super-symbol view of the coded signal that $\rho_{jk}(l)$ plays for the standard view of the signal. In fact, $\tilde{\rho}$, $\tilde{U}$, $\tilde{V}$ and $\tilde{W}$ can be related to ρ directly by substituting (7) into (11) through (14).

$$\tilde{\rho}_{jk}(l) = \rho_{jk}(2l+m_k-m_j) \tag{15a}$$

$$\tilde{U}_{jk}(l) = \rho_{jk}(2l+m_k-m_j) \tag{15b}$$

$$\tilde{V}_{jk}(l) = \rho_{jk}(2l+1+m_k-m_j) \tag{15c}$$

$$\tilde{W}_{jk}(l) = \rho_{jk}(2l-1+m_k-m_j) \tag{15d}$$

Note that $\tilde{\rho}_{jk}(l) = \tilde{U}_{jk}(l) = \tilde{V}_{jk}(l) = \tilde{W}_{jk}(l) = 0$ for $|l|>1$; this fact will play an important role in determining the proper state description of the system for the optimal sequence estimator. Some other useful properties of the correlation parameters are $\tilde{\rho}_{jk}(l) = \tilde{\rho}_{kj}(-l)$, $\tilde{U}_{jk}(l) = \tilde{U}_{kj}(-l)$ and $\tilde{W}_{jk}(l) = \tilde{V}_{kj}(-l)$.

Beginning with equation (8), note that by performing a modulo-K decomposition of the index i, namely $i = \alpha(i)K + \beta(i) - 1$, we can write

$$r(t) = \sum_{i=-M}^{M} \tilde{D}_{\beta(i)}(t-\alpha(i)T_b-\theta_{\beta(i)})\,\tilde{s}_{\beta(i)}(t-\alpha(i)T_b-\theta_{\beta(i)})\sqrt{E_{\beta(i)}} + n(t) \tag{16}$$

This assumes that the K users transmit $(2M+1)/K$ information bits each in the time interval of interest. We now further simplify the notation by defining the following terms,

$$E^i = E_{\beta(i)} \tag{17}$$

$$D_i^{(j)} = D_{\beta(i)}^{(j)}(\alpha(i)) \tag{18}$$

$$\tilde{\rho}_{im} = \tilde{\rho}_{\beta(i)\beta(m)}(\alpha(m)-\alpha(i)) \tag{19}$$

$$\tilde{U}_{im} = \tilde{U}_{\beta(i)\beta(m)}(\alpha(m)-\alpha(i)) \tag{20}$$

$$\tilde{V}_{im} = \tilde{V}_{\beta(i)\beta(m)}(\alpha(m)-\alpha(i)) \tag{21}$$

$$\tilde{W}_{im} = \tilde{W}_{\beta(i)\beta(m)}(\alpha(m)-\alpha(i)) \tag{22}$$

We have now laid the foundation for the derivation of the MLSE. This development will closely follow the derivation of the optimal MLSE in [7] and [8] for the uncoded ISI channel.

By expanding (16) with a Karhunen-Loeve expansion and letting the number of basis functions grow to infinity, we obtain the following waveform metric:

$$J_0(\overline{D}_M) = -\int_{-\infty}^{\infty} \Big(r(t) - \sum_{i=-M}^{M} D_i^{(1)} \phi_{1\beta(i)}(t-\alpha(i)T_b-\theta_{\beta(i)})\sqrt{E^i} + D_i^{(2)} \phi_{2\beta(i)}(t-\alpha(i)T_b-\theta_{\beta(i)})\sqrt{E^i} \Big)^2 dt \tag{23}$$

We next define:

$$r_i^{(1)} = r_{\beta(i)}^{(1)}(\alpha(i)) = \int_{-\infty}^{\infty} r(t)\phi_{1\beta(i)}(t-\alpha(i)T_b-\theta_{\beta(i)})dt \tag{24}$$

and

$$r_i^{(2)} = r_{\beta(i)}^{(2)}(\alpha(i)) = \int_{-\infty}^{\infty} r(t)\phi_{2\beta(i)}(t-\alpha(i)T_b-\theta_{\beta(i)})dt \tag{25}$$

which represent the outputs of a pair of matched filters or correlators for the basis functions for user $\beta(i)$ at time $t = (\alpha(i)+1)T_b + \theta_{\beta(i)}$. By expanding (23), and then collecting the appropriate terms we get the following metric.

$$\begin{aligned} J_0(\overline{D}_i) = J_0(\overline{D}_{i-1}) + 2\Big[& D_i^{(1)}\sqrt{E^i}\,\Big(r_i^{(1)} - \sum_{l=1}^{L}\{D_{i-l}^{(1)}\tilde{\rho}_{i\,i-l} + D_{i-l}^{(2)}\tilde{V}_{i\,i-l}\}\sqrt{E^{i-l}}\Big) \\ & + D_i^{(2)}\sqrt{E^i}\,\Big(r_i^{(2)} - \sum_{l=1}^{L}\{D_{i-l}^{(1)}\tilde{W}_{i\,i-l} + D_{i-l}^{(2)}\tilde{U}_{i\,i-l}\}\sqrt{E^{i-l}}\Big)\Big] \end{aligned} \tag{26}$$

where $\overline{D}_i$ represents the code sequence up to time interval i, and L is the smallest integer such that for every $L' > L$ we have $\tilde{\rho}_{jk}(\alpha(L')) = \tilde{U}_{jk}(\alpha(L')) = \tilde{V}_{jk}(\alpha(L')) = \tilde{W}_{jk}(\alpha(L')) = 0$. We have already seen that the correlation parameters are zero when $|\alpha(L')| > 1$, so $L = K-1$.

There are a number of important observations that can be made from the metric given in equation (26). First of all, the i^{th} stage metric depends only on the code symbols in the set

$$\Xi = \{D_i^{(1)},\, D_i^{(2)},\, D_{i-1}^{(1)} D_{i-1}^{(2)},\, \ldots,\, D_{i-K+1}^{(1)}\ D_{i-K+1}^{(2)}\}, \tag{27}$$

along with the matched filter outputs, $r_i^{(1)}$ and $r_i^{(2)}$, as well as the energies and correlations. It is possible to estimate the crosscorrelations using the local oscillators and code generators which are assumed to be synchronized to the K components of the incoming signal. We can also estimate the energies, $\{E^i\}$ by averaging the outputs of the matched filters for a number of bits. (The number over which they would be averaged would depend on the rate at which the relative strengths of the users is varying.)

For any user, k, the convolutional encoder defines a mapping rule from the input information symbol and present state of the encoder to the code bits, $D_k^{(1)}(j)$ and $D_k^{(2)}(j)$. If we define $h(\cdot)$ as the mapping rule from the input information symbol and state to the 4-ary super-code symbol, $\{D_k^{(1)}(j), D_k^{(2)}(j)\}$ then by substituting the information symbols that define the state of the encoder in for the state, the following expression may be written:

$$\{D_k^{(1)}(j), D_k^{(2)}(j)\} = h(I_k(j), I_k(j-1), \ldots, I_k(j-W+1)) \tag{28}$$

Thus, in this form, it is clear that the 4-ary super-code symbol depends on only W information symbols. Using this information, it is easy to redefine the set Ξ which was defined in equation (27) in terms of the information symbols which affect the i^{th} stage metric.

$$\tilde{\Xi} = \{I_i, \sigma_i\} \tag{29}$$

$$\sigma_i = \{I_{i-1}, I_{i-2}, \ldots, I_{i-WK+1}\} \tag{30}$$

where $I_i = I_{\beta(i)}(\alpha(i))$. Thus it is now apparent that the system may be described in terms of 2^{WK-1} states, since the information symbols are binary. Furthermore, the maximum likelihood sequence estimator can be implemented with a Viterbi algorithm operating on a trellis with 2^{WK-1} states and two branches per state. This trellis will be cyclically time-varying as in the uncoded case. [1] Furthermore, it is clear that this trellis reduces to the trellis derived in [1] when the constraint length of the code is one (uncoded transmission for each user). Obviously, the number of states in the MLSE grows very quickly with both the number of users in the system and the constraint length of the codes being used. In fact, for a simple 4-user case where each user uses a $W = 3$, or 4-state code, the MLSE requires a Viterbi algorithm operating on a trellis with 2048 states!

The time complexity per bit decoded for the MLSE is $TCB = O(2^{WK})$ since there are $2 \cdot 2^{WK-1}$ metrics which must be computed at each stage of the trellis and one information bit is decided at each stage. Note that for the case of $W = 1$, the TCB calculated in [1] is again obtained.

Now that the MLSE has been derived for the rate-1/2 case, it is straightforward to generalize to the case of rate-P/Q convolutional codes. The function $\tilde{D}_k(t)$ will again have to be constructed from a set of orthogonal basis functions. One reasonable choice would be a set of Q non-overlapping pulses of duration T_b/Q. Again, the functions $\{\tilde{s}_k(t)\}$ would be constructed from concatenations of Q versions of $s_k(t)$. The metric derivation could then proceed in the same fashion as in the rate-1/2 case. There are 2^P input hypotheses to test in each information bit period for each user, so the overall trellis will have 2^P branches per state. Furthermore, the state of the system will be specified by $(\kappa+P)(K-1)+\kappa$ information bits, so it will have $2^{\kappa K+PK-P}$ states, where $\kappa = \log_2 S$ and S is the number of states in the single user's encoder. This will result in a $TCB = O(2^{\kappa K+PK}/P)$.

Clearly the exponential dependence of the TCB on the number of users, the number of states in each of the user's codes and P makes the use of the optimal decoder prohibitive for a realistic system. It is, however, an important receiver because it represents the best that can be achieved in terms of sequence error probability, and it will provide a good baseline on which to judge the quality of suboptimal schemes. This receiver also raises the possibility of using a variety of sparse searching algorithms like a sequential decoder as was used in [5] for the uncoded case, or reduced state sequence estimation techniques like the one proposed in [4] for the uncoded MUI equalization problem or [9] for the combined equalization and decoding problem for single-user links suffering from ISI.

3.2 Performance of the Optimal Sequence Estimator

To illustrate the derivation of some performance bounds for the MLSE, we will again use the rate 1/2 code example. In this analysis we will follow fairly closely the analysis which appeared in [1] and [7]. In keeping with [1], we consider the decoding window to range from the index $-M$ to the index M. The goal of this section is to estimate the performance of the optimal sequence estimator by bounding the finite and infinite horizon error probabilities for the k^{th} user in the system, denoted $P_k^M(i)$ and $P_k = \lim_{M \to \infty} P_k^M(i)$.

We now consider the transmission of the sequence of super-code symbols, $\overline{D} = \{D_i^{(1)}, D_i^{(2)}\}_{i=-M}^{M}$, and a competing sequence $\overline{D}+2\overline{e}$ corresponding to the sequence $\{D_i^{(1)}+2e_i^{(1)}, D_i^{(2)}+2e_i^{(2)}\}_{i=-M}^{M}$ where $\overline{e} = \{e_i^{(1)}, e_i^{(2)}\}_{i=-M}^{M}$ is a sequence of error symbols. Each $e_i^{(j)}$ can take on values in the set $F = \{0, \pm 1\}$. Next, define the following sets:

$$B = \{\, \overline{e}: e_k^{(j)}(n) \varepsilon F,\ n = \alpha(-M),\ldots,\alpha(M),\ k=1,\ldots,K,\ j=1,2,\ e_k^{(j)}(n) \neq 0 \text{ for } \textit{some } n,k,j \,\} \tag{31}$$

$$A(\overline{D}) = \{\, \overline{e}: \overline{e} \varepsilon B,\ \overline{D}+2\overline{e} \varepsilon C \,\} \tag{32}$$

$$C = \{\, \overline{D}: \overline{D} \,\varepsilon\, h(\{\overline{I}\})\} \tag{33}$$

where $h(\cdot)$ is the mapping rule defined by the code from an information sequence, $\overline{I}$, to a sequence of super-code symbols, D, as in (28). Since this mapping rule is a one to one function, it has an inverse. If we define the information error sequence

$$\overline{\psi} = h^{-1}(\overline{D}+2\overline{e}) - \overline{I} \tag{34}$$

which is the information bit error sequence corresponding to $\overline{D}+2\overline{e}$ such that if $\overline{D} = h(\overline{I})$, then $\overline{D}+2\overline{e} = h(\overline{I}+\overline{\psi})$. This allows us to define

$$A_k^M(\overline{D},n) = \{\overline{e}: \overline{e} \varepsilon A(\overline{D}),\ \psi_k(n) \neq 0\} \tag{35}$$

so $A_k^M(\overline{D},n)$ is the set of admissible error sequences which affect the n^{th} information bit of the k^{th} user. From these definitions, it follows that the probability of error for the n^{th} bit of the k^{th} user is given by

$$P_k^M(n) = \sum_{\bar{D}\varepsilon C} P\left[\bigcup_{\bar{e}\varepsilon A_k^M(\bar{D},n)} \{J_0(\bar{D}+2\bar{e})>J_0(\bar{D}) \mid \bar{D}\ sent\ \}\right] \cdot P^M(\bar{D}\ sent) \tag{36}$$

As is the usual approach, we choose to bound (36) with a union bound.

$$P_k^M(n) \le \sum_{\bar{D}\varepsilon C} \ \sum_{\bar{e}\varepsilon A_k^M(\bar{D},n)} P(J_0(\bar{D}+2\bar{e})>J_0(\bar{D}) \mid \bar{D}\ sent) \cdot P^M(\bar{D}\ sent) \tag{37}$$

The event $J_0(\bar{D}+2\bar{e}) > J_0(\bar{D})$ may now be written by expanding equation (23) and substituting

$$r_i^{(1)} = r_{\beta(i)}^{(1)}(\alpha(i)) = \sum_{j=i-K+1}^{i+K-1} (D_j^{(1)}\sqrt{E^j}\,\tilde{\rho}_{ij} + D_j^{(2)}\sqrt{E^j}\,\tilde{V}_{ij}) + z_i^{(1)} \tag{38}$$

and

$$r_i^{(2)} = r_{\beta(i)}^{(2)}(\alpha(i)) = \sum_{j=i-K+1}^{i+K-1} (D_j^{(1)}\sqrt{E^j}\,\tilde{W}_{ij} + D_j^{(2)}\sqrt{E^j}\,\tilde{U}_{ij}) + z_i^{(2)} \tag{39}$$

for $r_i^{(1)}$ and $r_i^{(2)}$ respectively, where $z_i^{(1)}$ and $z_i^{(2)}$ are the noise statistics at the output of the matched filters for the basis functions $\phi_{1\beta(i)}$ and $\phi_{2\beta(i)}$ respectively for the interval $\alpha(i)$. After some algebra, the following expression for this event is obtained.

$$\begin{aligned} \Delta^2(\bar{e}) &= \sum_{i=-M}^{M} \sum_{m=-M}^{M} (e_i^{(1)}e_m^{(1)}\tilde{\rho}_{im} + e_i^{(2)}e_m^{(2)}\tilde{U}_{im} + e_i^{(1)}e_m^{(2)}\tilde{V}_{im} + e_i^{(2)}e_m^{(1)}\tilde{W}_{im})\,\sqrt{E^iE^m} \\ &< \sum_{i=-M}^{M} \sqrt{E^i}\,(e_i^{(1)}z_i^{(1)} + e_i^{(2)}z_i^{(2)}) \end{aligned} \tag{40}$$

The right side of equation (40) is a linear combination of Gaussian random variables, $z_i^{(1)}$ and $z_i^{(2)}$. We know that $E[z_i^{(1)}] = E[z_i^{(2)}] = 0$ and also that

$$E\left[\begin{pmatrix} z_i^{(1)} \\ z_i^{(2)} \end{pmatrix} (z_m^{(1)}\ z_m^{(2)})\right] = \frac{N_0}{2}\cdot\begin{bmatrix} \tilde{\rho}_{im} & \tilde{V}_{im} \\ \tilde{W}_{im} & \tilde{U}_{im} \end{bmatrix} \tag{41}$$

As a result, if we define γ to be the right side of equation (40), then it is not difficult to show that $E[\gamma]=0$ and $Var[\gamma]=\Delta^2(\bar{e})\cdot N_0/2$. Next, the two-sequence error probability, or the probability of the event given in equation (40), becomes the probability that the Gaussian random variable, γ, is larger than the threshold, $\Delta^2(\bar{e})$. We next define the following efficiency parameter for the pair of sequences separated by the code symbol error sequence, $\bar{e}$, as

$$\eta_k^M(\bar{e}) = \frac{\Delta^2(\bar{e})}{2E_k} = \frac{\Delta^2(\bar{e})}{E_{bk}} \tag{42}$$

where $E_{bk} = 2E_k$ is the energy per information bit for user k. This allows us to write

$$P(J_0(\bar{D}+2\bar{e}) > J_0(\bar{D}) \mid \bar{D}\ sent) = Q\left[\sqrt{\frac{2E_{bk}}{N_0}\cdot\eta_k^M(\bar{e})}\right] \tag{43}$$

so $\eta_k^M(\bar{e})$ may be viewed as an asymptotic efficiency relative to uncoded BPSK transmission for the k^{th} user for the pair of sequences $\bar{D}$ and $\bar{D}+2\bar{e}$. This can be shown to reduce to the form of the distance measure in [1] for the uncoded system.

In order to construct a lower bound on the probability of error for user k, we define the following minimum efficiency as

$$\eta_{k,min}^M(n) = \inf_{\bar{D}\varepsilon C}\ \inf_{\bar{e}\varepsilon A_k^M(\bar{D},n)} \eta_k^M(\bar{e}) \tag{44}$$

and so

$$P_k^M(n) \ge P[\eta_k^M(\bar{e}) = \eta_{k,min}^M(n)]\cdot Q\left[\sqrt{\frac{2E_{bk}}{N_0}\cdot\eta_{k,min}^M(n)}\right] \tag{45}$$

So we now have a lower bound expression for $P_k^M(n)$ given in (45) and an upper bound expression when (43) is substituted into equation (37).

To obtain bounds for the infinite horizon error probabilities we may conclude exactly as in [1] that the infinite horizon efficiencies $\eta_k(\overline{e})$ and $\eta_{k,min}$ are achieved by finite length error sequences. As a result, the infinite horizon error probability for the k^{th} user may be lower bounded by

$$P_k \geq P\left[\eta_k(\overline{e})=\eta_{k,min}\right]\cdot Q\left[\sqrt{\frac{2E_{bk}}{N_0}\cdot\eta_{k,min}}\right] \tag{46}$$

Similarly, by passing (37) to the limit as M approaches infinity

$$P_k \leq \sum_{\overline{D}\varepsilon C}\ \sum_{\overline{e}\varepsilon A_k(\overline{D},n)} P(\overline{D}\ sent)\cdot Q\left[\sqrt{\frac{2E_{bk}}{N_0}\cdot\eta_k(\overline{e})}\right], \tag{47}$$

where $A_k(\overline{D},n)=\lim_{M\to\infty} A_k^M(\overline{D},n)$.

In the high signal to noise ratio regime, the terms in (47) with the minimum efficiency will dominate the asymptotic behavior of the receiver. As a result, we will refer to the minimum efficiency, $\eta_{k,min}$ as the *asymptotic multiuser coding gain* for user k. The asymptotic multiuser coding gain is an efficiency parameter which is a measure of the energy gain or loss of the system relative to an uncoded BPSK system operating alone with an energy per information bit of E_{bk}.

In the limiting cases where there is only $K=1$ user in the system, or when there are K users in the system with orthogonal super-signature sequences, then $\eta_{k,min}$ is the coding gain of a single-user system operating with the same code. In the limiting case where the users do not employ coding, $\eta_{k,min}$ is equivalent to the asymptotic multiuser efficiency obtained in [1] for the optimal multiuser receiver for the uncoded system. Thus the asymptotic multiuser coding gain unifies the asymptotic coding gain and the asymptotic multiuser efficiency parameters. This parameter will also allow the study of near-far resistance for this receiver, which is the infimum of $\eta_{k,min}$ over the energies of all of the interfering users.

In general, the computation of $\eta_{k,min}$ is a difficult problem. In this paper, to provide some direct comparisons between the performance of the MLSE and the conventional receiver, we will use a computer simulation for some two-user cases. Figure 2 shows the results of a simulation of a two-user system where each user employs a 4-state rate 1/2 convolutional code. The resulting super-trellis used by the MLSE has 32 states. Figure 2 illustrates a very severe MUI environment where $\rho_{12}(0)=0.3$ and $\rho_{12}(-1)=0.3$. In this case, the MLSE is able to recoup almost all of the loss that the conventional decoder suffers when compared with the performance in the single-user environment. Also shown for comparison is a representative of the decision feedback approaches proposed in [11]. In Figure 3, the same 0.3 channel is simulated for a varying near-far energy ratio. This figure shows that the MLSE approach appears to be near-far resistant, although this has yet to be shown analytically. Again, for comparison, the conventional receiver and the best of the decision feedback approaches from [11] are shown.

It is worth noting that all of the performance analysis in this section has been based upon the metric for the case where each user in the system employs rate-1/2 convolutional codes. The expression for the distance and asymptotic multiuser coding gain will be more complicated in the general rate-P/Q code case, but the derivation procedure will be the same. Thus the work in this section is meant to illustrate the general procedure for the error analysis of the more complex general code rate case.

4. Conclusions

In this paper, trellis-based multiuser decoders were studied for links which use CDMA and convolutional codes. Two trellis-based receivers have been discussed. The first was a structure which attacked the MUI and the decoding operations separately. The second approach was an optimum sequence estimator for the problem. It was shown that the complexity of the MLSE depends exponentially on the number of users in the system, the number of states in each user's encoder and the number of input information bits, P. This high complexity points to the use of suboptimal approaches to attempt to attain high performance levels with a more reasonable complexity, such as reduced state sequence estimation approaches, sequential decoding approaches, linear approaches, [6], or multistage decision feedback approaches, [11].

5. References

[1] S. Verdu, "Minimum Probability of Error for Asynchronous Gaussian Multiple Access Channels," *IEEE Trans. Inform. Theory*, vol. IT-32, pp 85-96, Jan. 1986.

[2] S. Verdu, "Optimum Multiuser Asymptotic Efficiency," *IEEE Trans. on Comm.*, vol.34, no.9, May 1991, pp. 890-897.

[3] R. Lupas and S. Verdu, "Near-Far Resistance of Multiuser Detectors in Asynchronous Channels," *IEEE Trans. on Comm.*, vol.38, no.4, April 1990, pp 496-508.

[4] M.K. Varanasi, "Reduced State Sequence Detection for Asynchronous Gaussian Multiple-Access Channels", *Proceedings of ISIT*, San Antonio, Texas, January, 1993, pp. 42.

[5] Z. Xie, C.K. Rushforth, R.T. Short, "Multiuser Signal Detection Using Sequential Decoding," *IEEE Trans. on Comm.*, vol. 38, no. 5, May 1990, pp. 578-583.

[6] S.S.H. Wijayasuriya, G.H. Norton, J.P. McGeehan, "A Sliding Window Decorrelating Algorithm for DS-CDMA Receivers," *Electronics Letters*, vol. 28, no. 17, August, 1992, pp.1596-1598.

[7] G. Ungerboeck, "Adaptive Maximum-Likelihood Receiver for Carrier-Modulated Data-Transmission Systems," *IEEE Trans. on Comm.*, vol. 22, no. 5, May 1974, pp. 624-636.

[8] J.G. Proakis, *Digital Communications*, New York: McGraw-Hill, 1989.

[9] P.R. Chevillat, E. Eleftheriou, "Decoding of Trellis-Encoded Signals in the Presence of Intersymbol Interference and Noise," *IEEE Trans. on Comm.*, vol. 37, no. 7, July 1989, pp. 669-676.

[10] J. Hagenauer, P. Hoeher, "A Viterbi Algorithm with Soft-Decision Outputs and its Applications," *Proceedings of GLOBECOM '89*, Dallas, Texas, Nov. 1989, pp. 1680-1686.

[11] T.R. Giallorenzi, S.G. Wilson, "Multistage Decision Feedback and Trellis-Based Multiuser Receivers for Convolutionally Coded CDMA Systems," *Technical Report UVA/538341/EE93/102*, University of Virginia, May 1993.

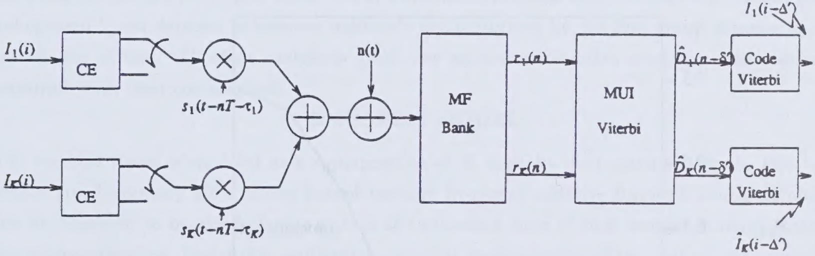

$I_1(i)$
CE
$s_1(t-nT-\tau_1)$
$I_K(i)$
CE
$s_K(t-nT-\tau_K)$
n(t)
MF
Bank
$r_1(n)$
$r_K(n)$
MUI
Viterbi
$\hat{D}_1(n-\delta')$
$\hat{D}_K(n-\delta')$
Code
Viterbi
Code
Viterbi
$\hat{I}_1(i-\Delta')$
$\hat{I}_K(i-\Delta')$

Figure 1a Partitioned Trellis-Based Equalizer/Decoder.

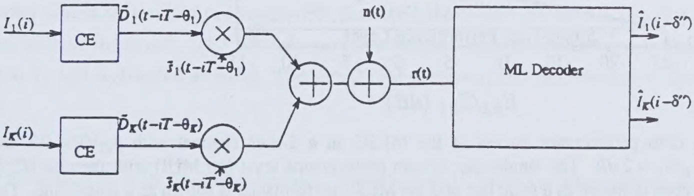

$I_1(i)$
CE
$\tilde{D}_1(t-iT-\theta_1)$
$\tilde{s}_1(t-iT-\theta_1)$
$I_K(i)$
CE
$\tilde{D}_K(t-iT-\theta_K)$
$\tilde{s}_K(t-iT-\theta_K)$
n(t)
r(t)
ML Decoder
$\hat{I}_1(i-\delta'')$
$\hat{I}_K(i-\delta'')$

Figure 1b Maximum Likelihood Sequence Estimator for the equivalent problem.

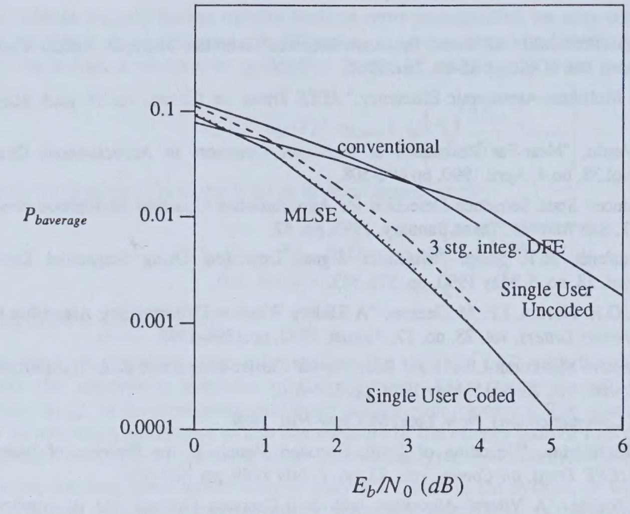


Figure 2 Performance curves of the MLSE (dotted line) for a 2-user channel with $\rho_{12}(0) = 0.3$ and $\rho_{12}(-1) = 0.3$ and equal energies. The solid lines show a single user system (no MUI) with and without the rate-1/2 4-state convolutional code. A three stage integrated decision feedback equalizer from [11] is also shown for comparison as a dashed line.

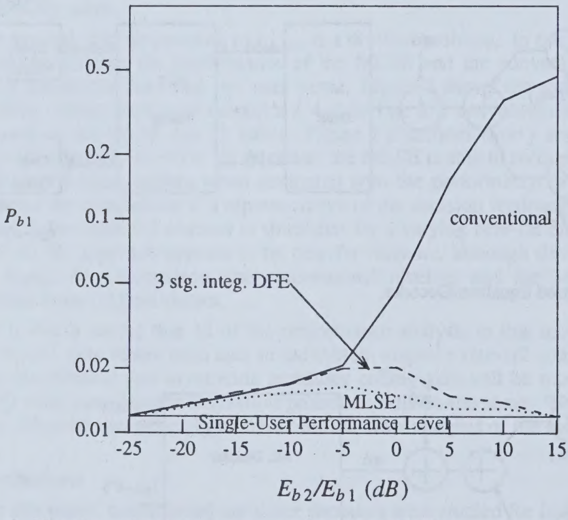


Figure 3 Near-far ratio performance curves of the MLSE on a 2-user channel with $\rho_{12}(0) = 0.3$ and $\rho_{12}(-1) = 0.3$ at $E_{b1}/N_0 = 2\ dB$. The single-user system performance level (no MUI) with the rate-1/2 4-state convolutional code is shown as a solid line and the MLSE performance is shown as a dotted line. The three stage integrated decision feedback equalizer from [11] is shown as a dashed line for comparison.

Group Detection for Synchronous CDMA Communication Over Frequency-Selective Fading Channels

MAHESH K. VARANASI†

Abstract: Two group detectors are introduced for the frequency-selective Rayleigh fading Code-Division Multiple-Access (FSRF-CDMA) channel. A group detector performs the joint detection of the information symbols transmitted by a group of users, and a bank of J group detectors, one for demodulating the information symbols of the users in each group of a J group partition of the K simultaneously transmitting users, constitutes a group detection scheme. While the optimum multiuser detector for the FSRF-CDMA channel has a complexity that is exponential in K, the two group detectors proposed have a complexity that is exponential only in the size of the corresponding group. The partition of users is a design parameter and hence group sizes can be chosen to satisfy a wide range of complexity constraints. A measure of asymptotic efficiency is introduced for the FSRF-CDMA channel. A lower bound on the asymptotic efficiency of the first group detector and an exact formula for the asymptotic efficiency of the second group detector are obtained. It is shown that larger group sizes result in higher performance which dictates partitions of users into groups of maximum possible size for given complexity constraints. The group detection approach is shown to result in a unifying solution to the complexity-performance trade-off problem in multiuser communication over the FSRF-CDMA channel. Both group detectors that correspond to the trivial partition of all users into one group of size K, are identical to the optimum multiuser detector. The particular case of the second group detector when the group size is equal to one yields a suboptimum linear detector that was recently proposed. This suboptimal linear detector is however uniformly out-performed by the first group detector specialized for a group size of one. All other partitions yield new multiuser detection strategies with performances commensurate with their complexities.

I. SYSTEM MODEL

The received signal is modeled as a superposition of K digitally modulated waveforms that arrive at the receiver simultaneously after having passed through frequency-selective Rayleigh fading channels. The channels are assumed to be *slowly fading* so that the coherence time of each channel is much greater than the data symbol duration. Under this assumption, accurate measurements of the channel characteristics can be obtained. Furthermore, the bandwidth assigned to each of the signature waveforms B_w is assumed to be much larger than the coherence bandwidth f_c of the channel so that the k^{th} channel can be represented as a tapped delay line with tap spacing $1/B_w$, and tap weight functions $\{c_{k,i}(t)\}_{i=0}^{L-1}$ for $k = 1, \cdots, K$. These fading parameters represent the channel state information and are modeled as zero-mean complex-valued stationary Gaussian random processes. The number of resolvable paths L is $[B_w/f_c] + 1$, where $1/f_c$ is the channel multipath spread. The k^{th} user complex lowpass signature waveform is time-limited to a symbol duration T, and is denoted as $u_k(t)$. The signal that arrives at the receiver due to the isolated transmission of the signature waveform of the k^{th} user is given as

---

† The author is with the Department of Electrical and Computer Engineering, University of Colorado, Boulder, CO-80309. This work was supported in part by NSF Grant NCR-9206327

$$h_k(t) = \sqrt{w_k}\sum_{i=0}^{L-1} c_{k,i}(t)u_k(t - \frac{i}{B_w}) = \sqrt{w_k}\mathbf{u}_k^T(t)\mathbf{c}_k(t) \qquad (1),$$

where the k^{th} fading channel vector is defined as $\mathbf{c}_k(t) = [c_{k,0}(t), \cdots, c_{k,L-1}(t)]^T$, and the k^{th} user's signature waveform and its time-translates are defined by the vector

$$\mathbf{u}_k(t) = [u_k(t)\ u_k(t - 1/B_w)\ \cdots\ u_k(t-(L-1)/B_w)]^T \ .$$

When a user transmits a sequence of information symbols, there is intersymbol interference and the optimal detector in the single-user environment is the maximum likelihood sequence detector that can be implemented via a dynamic programming algorithm. With multiple users there is intersymbol interference between adjacent symbols of the same user as well as from those of other users and a multiuser maximum likelihood sequence detector is optimal in this case [ZvB93]. However, there are several applications where the symbol duration is much larger than the multipath spread, $T >> 1/f_c$, so that $L/B_w << T$ and the intersymbol interference (ISI) due to channel dispersion is ignored (cf. [Pro89] [ZvB92]). Alternatively, ISI can be entirely eliminated by making symbol-by-symbol decisions based only on the received signal over time-intervals where ISI is absent. The small loss of optimality is traded for the simplicity of the resulting symbol-by-symbol decision rules. The representation of the received signal over the zeroth time-interval is given by

$$r(t) = \sum_{k=1}^{K} b_k h_k(t) + n(t) = \mathbf{h}^T(t)\mathbf{b} + n(t) \ , \qquad L/B_w \le t \le T \qquad (2)$$

where $\mathbf{b} = [b_1, \cdots, b_K]^T$ with b_k denoting the k^{th} user's information symbol and the signal vector $\mathbf{h}(t) = [h_1(t), \cdots, h_K(t)]^T$. The additive white Gaussian noise process $n(t)$ has a power spectral density of height N_0. The symbol b_k belongs to F_k, an M-ary PAM constellation of the k^{th} user.

A succinct description of the problem is facilitated by using the following block matrix notation. The KL-length column vector of signals obtained by stacking the vectors $\{\mathbf{u}_k(t)\}_{k=1}^{K}$ is denoted as $\mathbf{u}(t)$. Similarly, the KL-length multiuser multipath channel vector $\mathbf{c}$ is defined as $\mathbf{c}(t) = [\mathbf{c}_1^T(t), \cdots, \mathbf{c}_K^T(t)]^T$. The $KL \times KL$ signature signal correlation matrix is defined as

$$\mathcal{R} = \int_{L/B_w}^{T} \mathbf{u}^*(t)\mathbf{u}^T(t)dt = \begin{bmatrix} \mathbf{R}_{11} & \mathbf{R}_{12} & \cdots & \mathbf{R}_{1K} \\ \mathbf{R}_{21} & \mathbf{R}_{22} & \cdots & \mathbf{R}_{2K} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{R}_{K1} & \mathbf{R}_{K2} & \cdots & \mathbf{R}_{KK} \end{bmatrix} , \qquad (3)$$

with the superscripts $*$ and T denote complex conjugation and matrix transposition, respectively. The matrix $\mathbf{R}_{kl}$ is the correlation matrix between the L-dimensional signal vectors $\mathbf{u}_k(t)$ and $\mathbf{u}_l(t)$ defined as $\mathbf{R}_{kl} = \int_{L/B_w}^{T} \mathbf{u}_k^*(t)\mathbf{u}_l^T(t)dt$. The signature signals are normalized and all their time-translates are assumed to be linearly independent so that $\mathcal{R}$ is positive definite with unit diagonal elements. A diagonal matrix $\mathbf{W}$ is defined as $diag\{w_1, \cdots, w_K\}$ where w_k is the k^{th} user's average received power defined as the expected value of $\int_{L/B_w}^{T} |h_k(t)|^2 dt$. It is also convenient to define the $KL \times K$ multipath multiuser channel state information matrix $\mathcal{C}$ as

$$\mathcal{C} = \begin{bmatrix} \mathbf{c}_1(t) & 0 & \cdots & 0 \\ 0 & \mathbf{c}_2(t) & \cdots & 0 \\ & & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{c}_K(t) \end{bmatrix}. \tag{4}$$

II. OPTIMUM DETECTION AND SUBOPTIMUM LINEAR DETECTION

A. The Optimal Detector

In this section, we consider centralized coherent multiuser detection for the nominal FSRF-CDMA channel where the fading channel matrix is assumed to be constant over the observed interval and known precisely at the receiver. All signature waveforms and their average received powers are also known at the receiver. The optimum multiuser detection problem under the given assumptions is a simple M^K-ary hypothesis testing problem. Since the additive noise is white and Gaussian and all the realizations of $\mathbf{b}$ are equiprobable, the optimum multiuser decision rule that achieves the minimum probability of error of the joint decision on $\mathbf{b}$, selects $\hat{\mathbf{b}}$ according to the minimum energy realization of the noise process. The optimum multiuser detector can be shown to be

$$\hat{\mathbf{b}} = \arg\max_{\mathbf{s}\in F} 2\mathrm{Re}\{\mathbf{q}^{*\mathrm{T}}\mathcal{C}\mathbf{W}^{1/2}\mathbf{s}\} - \mathbf{s}^{*\mathrm{T}}\mathbf{W}^{1/2}\mathcal{C}^{*T}\mathcal{R}\mathcal{C}\mathbf{W}^{1/2}\mathbf{s}\,, \tag{5}$$

where $\mathbf{q}$ is the output of a bank of time-invariant matched filters matched to each of the signature signals and their time-dispersed translates and is defined as

$$\mathbf{q} = \int_{L/B_w}^{T} r(t)\mathbf{u}^*(t)\,dt\,. \tag{6}$$

The combinatorial maximization in (5) is NP-hard in K. The time complexity per symbol (TCS) is $O(M^K/K)$ which is much too high for all but small values of K. The design and analysis of low-complexity, suboptimum multiuser detectors is therefore the key problem.

B. The Conventional Detector

The single-user RAKE receiver [Pro89] has been extensively analyzed over multiuser channels in recent spread spectrum literature. We refer to this receiver as the conventional detector. It optimally combines the multiple paths at the receiver under the assumption that the interfering users are absent. For BPSK modulation, the k^{th} user's decision is given by

$$\hat{b}_k = \mathrm{sgn}\left[\mathrm{Re}(\mathbf{c}_k^{*T}\mathbf{q}_k)\right], \tag{7}$$

where $\mathbf{q}_k$ is the k^{th} L-dimensional vector of $\mathbf{q}$. When employed over multiuser channels, the conventional RAKE detector is far from being optimal. An asymptotic analysis of the performance of the RAKE detector in [ZvB93] has uncovered a severe error-rate bottoming phenomenon for high average SNR due to multiple-access interference.

C. A Suboptimum Linear Detector

A suboptimal linear multiuser detector for the FSRF-CDMA channel has been recently proposed in [ZvB92]. The matched filter outputs $\mathbf{q}$ are linearly transformed according to the decorrelating linear transformation $\mathcal{Q} = \mathcal{R}^{-1}$, following which the L outputs corresponding to the k^{th} user are combined to maximize the signal-to-noise ratio of the resulting decision statistic. This detector for the k^{th} user is denoted as $\hat{\phi}_k$ and is defined by

$$\hat{\phi}_k : \qquad \hat{b}_k = \operatorname{sgn}\left[\operatorname{Re}(\mathbf{c}_k^{*T}\mathbf{Q}_{kk}^{-1}(\mathcal{Q}\mathbf{q})_k)\right], \tag{8}$$

where $(\mathcal{Q}\mathbf{q})_k$ is the k^{th} L-dimensional vector in $\mathcal{Q}\mathbf{q}$, and $\mathbf{Q}_{kk}$ is the $(k,k)^{th}$ block of the inverse correlation matrix $\mathcal{Q} = \mathcal{R}^{-1}$. This suboptimal detector has a bit error rate that is invariant to interfering signal powers and has the advantage of requiring only the desired user's channel state information. It has been shown to perform extremely well in the low bandwidth efficiency examples considered in [ZvB92]. For medium-to-high bandwidth efficiencies however, this detector suffers significant performance degradations relative to the optimum detector. This behaviour of the detector is later revealed in this paper.

III. GROUP DETECTION I

The concept of group detection was introduced in the context of the synchronous Gaussian CDMA channel in [Var92]. It is a natural strategy for multiuser detection where the K users are partitioned into J disjoint groups and are demodulated by a bank of J group detectors, one for each group in the partition. Each group detector performs the joint detection of symbols in the corresponding group.

Notation: let $\Omega = \{1, \cdots, K\}$, and $G, H \subseteq \Omega$. Denote the complement of G with respect to Ω as $\bar{G}$. For any $\mathbf{x} \in \mathcal{C}^K$, $\mathbf{x}_G \in \mathcal{C}^{|G|}$ is obtained from $\mathbf{x}$ by striking out $x_k \; \forall \; k \in \bar{G}$. For any $X \in \mathcal{C}^{K \times K}$, $X_{GH} \in \mathcal{C}^{|G| \times |H|}$ is obtained from X by striking out the i^{th} rows and the j^{th} columns $\forall \; i \in \bar{G}$ and $\forall j \in \bar{H}$.

A. Derivation

When all users are to be demodulated at a central receiver or in point-to-multipoint applications where the fading channel coefficients of all the users are the same, the channel state information matrix $\mathcal{C}$ is known at the receiver. Consequently, the multiuser maximal ratio combiner (MMRC) for $\mathbf{b}$ is $\mathcal{C}^{*T}\mathbf{q} + \mathcal{C}^T\mathbf{q}^*$. A statistical description of this minimal sufficient statistic, conditioned on the fading channel matrix $\mathcal{C}$, reveals an equivalence between the conditional FSRF-CDMA channel and a time-varying K-user additive Gaussian CDMA channel. Consequently, a real-arithmetic version of the group detector proposed in [Var92] can be derived for the FSRF-CDMA channel. In considering the demodulation of $\mathbf{b}_G$, it is assumed that the average signal powers of all interfering users in the set $\bar{G}$ are unknown, and a generalized likelihood ratio test is obtained as

$$\phi_G : \qquad \hat{\mathbf{b}}_G = \arg \max_{\mathbf{s}_G \in F_G} \{2 \;(\mathbf{s}_G^T \mathbf{W}_{GG}^{1/2}\mathbf{P}_{GG}^{-1}[\mathbf{P}(\mathcal{C}^{*T}\mathbf{q} + \mathcal{C}^T\mathbf{q}^*)]_G) - \mathbf{s}_G^T\mathbf{W}_{GG}^{1/2}\mathbf{P}_{GG}^{-1}\mathbf{W}_{GG}^{1/2}\mathbf{s}_G\}, \tag{9}$$

where we define $\mathbf{P}$ to be the matrix $(\mathcal{C}^{*T}\mathcal{R}\mathcal{C} + \mathcal{C}^T\mathcal{R}^*\mathcal{C}^*)^{-1}$.

The implementation of the group detector requires the combinatorial maximization in (9) which is NP-hard in the size of the group $|G|$. Fortunately, $|G|$ is a *design parameter.* In practice, if complexity considerations allow the implementation of an optimum detector for upto N users, the set of all active users has to be partitioned into groups of size no greater than N each. Note that the special case of the group detector where $G = \Omega$ coincides with the maximum likelihood detector which has the highest TCS of $O(M^K/K)$. Consider the special case of (9) where $G = \{k\}$ and $F_k = \{+1, -1\}$:

$$\phi_k : \qquad \hat{b}_k = \operatorname{sgn}\left[[\mathbf{P}(\mathcal{C}^{*T}\mathbf{q} + \mathcal{C}^T\mathbf{q}^*)]_k) \right] . \tag{10}$$

An efficient implementation of this detector can be achieved by a single time-varying matched filter [Var93b].

B. Asymptotic Efficiency

Let the average bit error probability of a detector ϕ for the k^{th} user in a FSRF-CDMA channel be denoted as $P_k(\mathbf{w}, N_0, \phi)$, where $\mathbf{w}$ is the vector of average received powers of the K users. Let $P_k^{SU}(w, N_0)$ be the average bit-error rate of the single-user RAKE receiver operating in the k^{th} single-user FSRF channel where the k^{th} user transmits in isolation and whose signal at the receiver has average power equal to w.

Definition: The *asymptotic efficiency* of a detector ϕ operating in the FSRF-CDMA channel is formally defined as

$$\eta_k(\phi) = \sup\left\{ 0 \leq r \leq 1; \ \lim_{N_0 \to 0} \frac{P_k(\mathbf{w}, N_0, \phi)}{P_{k,SU}(rw_k, N_0)} \leq 1 \right\} . \tag{11}$$

This measure captures the loss in bit-error rate (BER) performance of a detector in the high SNR region where the dominant degradation in performance is due to the presence of interfering users rather than additive noise. In particular, a detector with a non-zero value of asymptotic efficiency $\eta_k(\phi)$ for user k operating in high SNR conditions has a BER performance that rivals the BER of the single-user RAKE receiver designed for the k^{th} single user FSRF channel with average received power $w_k\eta_k(\phi)$.

An exact expression for the average bit-error probability of the single-user RAKE receiver in the k^{th} single-user channel is available in the literature on single-user communication in fading channels [SBS66]:

$$P_{k,SU}(w_k, N_0) = \sum_{l=1}^{L} \frac{\alpha_l}{2}\left[1 - \sqrt{\frac{w_k\zeta_l}{N_0 + w_k\zeta_l}} \right] , \tag{12}$$

where $\{\zeta_l\}$ are the non-zero eigenvalues (assumed distinct) of the matrix $\mathcal{K}_{kk}\mathcal{R}_{kk}$ with $\mathcal{K}$ being the covariance matrix of the fading channel parameter vector $\mathbf{c}$ so that $\mathcal{K}_{kk}$ is the covariance matrix of $\mathbf{c}_k$ and the coefficients $\{\alpha_l\}$ are defined as the residues

$$\alpha_l = \prod_{i=1, i\neq l}^{L} \frac{\zeta_l}{\{\zeta_l - \zeta_i\}} .$$

Notation: For any block vector $\mathbf{x} \in \mathcal{C}^{MK}$, $\mathbf{x}_G \in \mathcal{C}^{M|G|}$ is obtained from $\mathbf{x}$ by striking out the M length vectors $\mathbf{x}_k$ from $\mathbf{x}$ $\forall\ k \in \bar{G}$. For any $K \times K$ block matrix with matrix entries of size $M \times N$, i.e., for $\mathcal{X} \in \mathcal{C}^{MK \times NK}$, we define $\mathcal{X}_{GH} \in \mathcal{C}^{M|G| \times N|H|}$ as the block matrix obtained from $\mathcal{X}$ by striking out the i^{th} block rows and the j^{th} block columns $\forall\ i \in \bar{G}$ and $\forall j \in \bar{H}$.

Proposition 1 *The asymptotic efficiency of any user $k \in G$ employing the group detector ϕ_G is lower bounded as*

$$\eta_k(\phi_G) \geq \left(\frac{det\left[(\mathcal{Q}_{GG}^{-1})_{i_k i_k} \right]}{det(\mathcal{R}_{kk})} \right)^{\frac{1}{L}} , \tag{13}$$

where i_k denotes the index of user k in G.

Proof: Following [Var92], it can be shown that the union bound for the conditional bit error probability for user k employing the group detector ϕ_G is given as

$$P_k(\mathbf{w}, N_0, \phi_G/\mathcal{C}) \leq \sum_{\delta \in E_k^G} 2^{-(w(\delta)-1)} Q\left(\sqrt{\frac{2\delta^T \mathbf{W}_{GG}^{1/2} \mathbf{P}_{GG}^{-1} \mathbf{W}_{GG}^{1/2} \delta}{N_0}}\right), \tag{14}$$

where the set of G error sequences $E_k^G = \{\delta \in \{-1, 0, 1\}^{|G|};\ \delta_{i_k} = +1\}$ with $w(\delta)$ denoting the weight of the error sequence δ. An upper bound on the average error probability can be obtained by taking the expected value of the union bound. Unfortunately, the conditional pair-wise error probabilities in the summation of the union bound depend on $\mathcal{C}$ in a complicated way. The exact computation of the expected value of the union bound (when $G \neq \Omega$) therefore appears to be intractable. However, it can be shown that the conditional pair-wise error probabilities admit an upper bound given as

$$Q\left(\sqrt{\frac{2\delta^T \mathbf{W}_{GG}^{1/2} \mathbf{P}_{GG}^{-1} \mathbf{W}_{GG}^{1/2} \delta}{N_0}}\right) \leq Q\left(\sqrt{\frac{\delta^T \mathbf{W}_{GG}^{1/2} \mathcal{C}_G^{*T} \mathcal{Q}_{GG}^{-1} \mathcal{C}_G \mathbf{W}_{GG}^{1/2} \delta}{N_0}}\right). \tag{15}$$

Equality in the above inequality holds when $G = \Omega$. Let us rewrite $\mathcal{C}_G \mathbf{W}_{GG}^{1/2} \delta_G$ as $\mathcal{W}_{GG}^{1/2} \mathcal{D}_{GG} \mathbf{c}_G$ with $\mathcal{D}$ and $\mathcal{W}$ defined as block diagonal matrices $\mathcal{D} = \mathrm{diag}\{\mathbf{D}_1, \cdots, \mathbf{D}_K\}$ and $\mathcal{W} = \mathrm{diag}\{\mathbf{W}_1, \cdots, \mathbf{W}_K\}$, where $\mathbf{D}_k = \delta_k \mathbf{I}_L$ and $\mathbf{W}_k = w_k \mathbf{I}_L$ with $\mathbf{I}_L$ denoting a $L \times L$ identity matrix, and define the matrix $\mathcal{S}_{GG} = \mathcal{D}_{GG} \mathcal{W}_{GG}^{1/2} \mathcal{Q}_{GG}^{-1} \mathcal{W}_{GG}^{1/2} \mathcal{D}_{GG}$. Substituting the resulting expression for the upper bound into the conditional union bound in (14) and taking expected values, we obtain the union upper bound for the average bit error rate as

$$\begin{aligned} P_k(\mathbf{w}, N_0, \phi_G) &:= E_{\mathcal{C}}(P_k(\mathbf{w}, N_0, \phi_G/\mathcal{C})) \\ &\leq \sum_{\delta \in E_k^G} 2^{-(w(\delta)-1)} E_{\mathbf{c}_G}\left[Q\left(\sqrt{\frac{\mathbf{c}_G^{*T} \mathcal{S}_{GG} \mathbf{c}_G}{N_0}}\right)\right] \end{aligned} \tag{16}.$$

The expected value of the Q-function of a quadratic form in zero-mean Gaussian random vectors can be obtained in closed form (cf. [SBS66]). In particular,

$$\begin{aligned} \hat{P}_k(\mathbf{w}, N_0, \phi_G, \delta) &:= E_{\mathbf{c}_G}\left[Q\left(\sqrt{\frac{\mathbf{c}_G^{*T} \mathcal{S}_{GG} \mathbf{c}_G}{N_0}}\right)\right] \\ &= \sum_{l=1}^{w(\delta)L} \frac{\beta_l(\delta)}{2}\left[1 - \sqrt{\frac{\lambda_l(\delta)}{N_0 + \lambda_l(\delta)}}\right], \end{aligned} \tag{17}$$

where $\{\lambda_l(\delta)\}$ are the non-zero eigenvalues (assumed distinct) of the matrix $\mathcal{K}_{GG}\mathcal{S}_{GG}$ with $\mathcal{K}$ being the covariance matrix of the fading channel parameter vector $\mathbf{c}$ and assumed to be of full-rank, and the coefficients $\{\beta_l(\delta)\}$ are defined as the residues

$$\beta_l(\delta) = \prod_{i=1, i \neq l}^{w(\delta)L} \frac{\lambda_l(\delta)}{\{\lambda_l(\delta) - \lambda_i(\delta)\}}. \tag{18}$$

It can be shown that $\hat{P}_k^{(n)}(\mathbf{w}, N_0, \phi_G, \delta)$, the n^{th} derivative of $\hat{P}_k(\mathbf{w}, N_0, \phi_G, \delta)$, when evaluated at $N_0 = 0$ is given as

$$\hat{P}_k^{(n)}(\mathbf{w}, 0, \phi_G, \delta) = \begin{cases} 0 & \text{if } n < Lw(\delta); \\ f_{Lw(\delta)}/\det(\mathcal{K}_{GG}(\delta)\mathcal{S}_{GG}(\delta)) & \text{if } n = Lw(\delta). \end{cases} \tag{19}$$

where $\mathcal{K}_{GG}(\delta)$ and $\mathcal{S}_{GG}(\delta)$ are obtained by striking out the block rows and block columns of the block matrices $\mathcal{K}_{GG}$ and $\mathcal{S}_{GG}$ corresponding to the indices of the components of δ that are equal to zero, respectively. Further, $f_{w(\delta)L}$ is the $(w(\delta)L)^{th}$ element of the sequence $f_m = (2m-1)(2m-3)...3.1/2^m$. Similarly, $P^{(n)}_{k,SU}(rw_k, N_0)$, the n^{th} derivative of $P_{k,SU}(rw_k, N_0)$, when evaluated at $N_0 = 0$ is given by

$$P^{(n)}_{k,SU}(rw_k, 0) = \begin{cases} 0 & \text{if } n < L \; ; \\ f_L/\det(rw_k\mathcal{K}_{kk}\mathcal{R}_{kk}) & \text{if } n = L \; . \end{cases} \tag{20}$$

As a consequence of (19) and (20), we have

$$\lim_{N_0 \to 0} \frac{\hat{P}_k(\mathbf{w}, N_0, \phi_G, \delta)}{P_{k,SU}(rw_k, N_0)} = \begin{cases} 0 & \text{if } w(\delta) > 1 \; ; \\ \det(r\mathcal{R}_{kk})/\det\left([\mathcal{Q}^{-1}_{GG}]_{i_k i_k}\right) & \text{if } w(\delta) = 1. \end{cases} \tag{21}$$

The lower bound of the proposition is now obtained by substituting the union upper bound in (16) in place of the average bit error rate $P_k(\mathbf{w}, N_0, \phi_G)$ in the definition of asymptotic efficiency, and using the dominance of the unity weight error sequence implied by the above equation. •

IV. GROUP DETECTION II

Consider the FSRF CDMA channel where the channels through which the various signals travel are distinct as in multipoint-to-point communication applications. Suppose further that the receiver is required to only demodulate the information transmitted by a subset of the active transmitters. Consider the scenario where, for implementational simplicity, the fading parameters corresponding to the fading channels of interfering users are not estimated and tracked at the receiver. In such cases the group detector ϕ_G cannot be implemented. A second group detection strategy $\hat{\phi}_G$ that does not require the interfering channel state information is defined below:

$$\hat{\phi}_G : \qquad \hat{\mathbf{b}}_G = \arg \max_{\mathbf{s}_G \in F_G} \left\{ 2\mathrm{Re}(\mathbf{s}^{*T}_G \mathbf{W}^{1/2}_{GG} \mathcal{C}^{*T}_G \mathcal{Q}^{-1}_{GG} [\mathcal{Q}\mathbf{q}]_G) - \mathbf{s}^{*\mathrm{T}}_G \mathbf{W}^{1/2}_{GG} \mathcal{C}^{*T}_G \mathcal{Q}^{-1}_{GG} \mathcal{C}_G \mathbf{W}^{1/2}_{GG} \mathbf{s}_G \right\} . \tag{22}$$

A formal derivation based on a generalized likelihood ratio criterion and a vector space interpretation of the group detector in (22) can be found in [Var93b]. The combinatorial maximization in (22) is found to be NP-hard in $|G|$. The special case where $G = \Omega$ yields the most complex group detector that in fact coincides with the maximum likelihood detector. From the consideration of the combinatorial maximizations involved in the implemetation of the two group detectors, both ϕ_G and $\hat{\phi}_G$ have the same complexity. There are however differences in the implementation of these two detectors which are detailed in [Var93b].

Consider the special case $G = \{k\}$, where the group detector $\hat{\phi}_G$ reduces to

$$\hat{\phi}_k : \qquad \hat{b}_k = \arg \max_{s_k \in F_k} \left\{ 2\mathrm{Re}(s^*_k \sqrt{w_k} \mathbf{c}^{*T}_k \mathcal{Q}^{-1}_{kk} [\mathcal{Q}\mathbf{q}]_k) - |s_k|^2 w_k \mathbf{c}^{*T}_k \mathcal{Q}^{-1}_{kk} \mathbf{c}_k \right\} . \tag{23}$$

The significance of this detector is that it is computationally the most efficient, lending itself to decentralized implemetation. The received signal is simply processed by L time-invariant matched filters matched to the complex conjugate of the signal vector $\mathcal{Q}^{-1}_{kk}[\mathcal{Q}\mathbf{u}^*]_k$ whose outputs are passed on to a single-user MRC followed by a single-user QAM decision device. Moreover, the L time-invariant decorrelating matched filter impulse responses can be pre-computed. No interfering channel state information is required by this detector either.

The complexity of this decorrelating scheme is therefore the same that of a single-user RAKE receiver. For MPSK modulation, $\hat{\phi}_k$ does not require the average received power of the desired signal. For the further special case where $F_k = \{+1, -1\}$ the group detector reduces to the linear suboptimal detector given by $\hat{\phi}_k$ of (8). The proofs of the following proposition and corollary can be found in [Var93b].

Proposition 2 *The asymptotic efficiency of any user $k \in G$ employing the group detector $\hat{\phi}_G$ is given as*

$$\eta_k(\hat{\phi}_G) = \left(\frac{\det\left[(\mathcal{Q}_{GG}^{-1})_{i_k i_k} \right]}{\det \mathcal{R}_{kk}} \right)^{\frac{1}{\mathcal{L}}} . \tag{24}$$

Corollary 1 *Suppose that $G_1, G_2 \subseteq \{1, \ldots, K\}$ are such that $k \in G_2 \subseteq G_1$. Then the asymptotic efficiency of the group G_1 detector $\hat{\phi}_{G_1}$ is greater than that of the group G_2 detector $\hat{\phi}_{G_2}$.*

Note that the result of the corollary implies that the asymptotic efficiency of each user employing a group detection scheme for any partition of the set of K users is uniformly lower bounded by that of the bank of linear suboptimum detectors, i.e.,

$$\eta_k(\hat{\phi}_G) \geq \left(\frac{\det\left[(\mathcal{Q}_{kk}^{-1}) \right]}{\det \mathcal{R}_{kk}} \right)^{\frac{1}{\mathcal{L}}} = \eta_k(\hat{\phi}_k)$$

VII. NUMERICAL RESULTS AND CONCLUSIONS

In this section, we illustrate the analytical results obtained in this paper by considering a six-user CDMA system where the signature signals used are direct-sequence spread spectrum signals each of length 31. We consider an example where the number of resolvable paths for the first three users is 4 and there are 5 paths associated with the rest of the signals. Although the results of this paper were developed for the case where the number of paths for all the signal are the same, they can be easily extended to the case of unequal number of paths. Consequently, there are 27 signals, each derived from the six spread-spectrum signals of length 31 as the appropriate 27-length subsequences in the interval corresponding to zero ISI. This corresponds to a bandwidth efficiency factor of 1 which is considerably higher than that of any previously considered examples. It was noted in [ZvB92] that the suboptimum linear detector proposed therein performs near-optimally and alleviates the severe shortcomings of the conventional single-user RAKE receiver. However, this conclusion was based on examples corresponding to low bandwidth efficiency factors. We demonstrate that medium-high bandwidth efficiency CDMA communication over FSRF channels tells a different story.

Figure 1 depicts the asymptotic efficiencies of the group detector $\hat{\phi}_G$ for user 1 as the group size grows from 1 to 6. By the result of Proposition 1, these also represent *lower* bounds on the asymptotic efficiencies of the group detector ϕ_G. A group of size $|G|$ includes users indexed from 1 to $|G|$. The asymptotic efficiency of the suboptimum linear detector (when $G = \{1\}$) is found to be 0.051 which implies that this detector suffers an approximately 13 dB power penalty relative to single-user RAKE receiver for the single-user channel, whereas the maximum likelihood detector ($G = \Omega$) has unit asymptotic efficiency and it therefore rivals single user performance for high SNR values. On the other hand the group detectors with group sizes 2 and

3 have power penalties of approximately 6.2 dB and 4.0 dB thus improving upon the suboptimum linear detector performance by approximately 6.8 dB and 9 dB, respectively. Much of the loss is therefore alleviated by these detectors for a modest increase in complexity.

In the next example, we illustrate the union bounds on the average BER of the group detector $\hat{\phi}_G$ (which are also the union upper bounds on the average BER of the ϕ_G detector, although it is unclear just how tight these bounds are for the latter case (cf. [Var93b]). The system parameters are the same as in Figure 1. The received signal powers are all equal and the covariance matrix of the fading parameters $\mathcal{K}$ is assumed to be diagonal and is derived from typical exponential multipath intensity profiles (cf. [Pro89] [ZvB93]). The union bounds are shown for user 1 for each of the group detectors corressponding to group sizes that increase from 1 to 6 as specified before. The single-user RAKE receiver performance over the single-user channel is also included. The single-user BER serves as the absolute lower bound on the performance of any detector in the multiuser channel including the optimum detector. Notice from Figure 2, the near coincidence of the union bound on the BER of the optimum detector with the single-user BER. This testifies to the tightness of the union bound and more significantly, since the asymptotic efficiency of the optimum detector is equal to unity, it validates the faithfulness of the asymptotic efficiency to bit-error rate for signal-to-noise ratios as low as 6-8 dB even for the our example with unit bandwidth efficiency. The union bound for the suboptimum linear detector is in fact its exact bit-error rate and it can be seen that relative to the single-user or the optimum detector BER, the suboptimum linear detector incurs a 13 dB power penalty, a result that was predicted accurately by the asymptotic efficiency analysis. Similarly, it can be verified that the bounds on the BER of the group detectors for group sizes 2 and 3 alleviate this penalty by nearly 7dB and 9 dB which is again commensurate with the asymptotic efficiency analysis.

The examples considered make a persuasive case for the use of group detection strategies introduced when bandwidth efficient CDMA communication over FSRF channels is required. A clear trade-off between performance and complexity is demonstrated in this work whereby suitable partitions of the users can be made to best meet performance-complexity specifications.

References

[Pro89] J. G. Proakis, "*Digital Communications*," Mc-Graw Hill, NY, 1989.

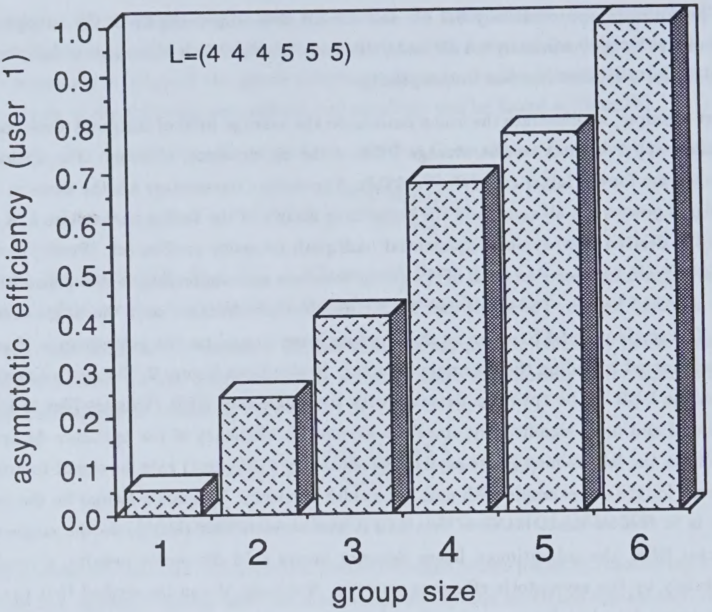
[SBS66] M. Schwartz, W. R. Bennett and S. Stein, *Communication Systems and Techniques*," Inter-University Electronics Series, Vol-4, McGraw-Hill Book Company, 1966.

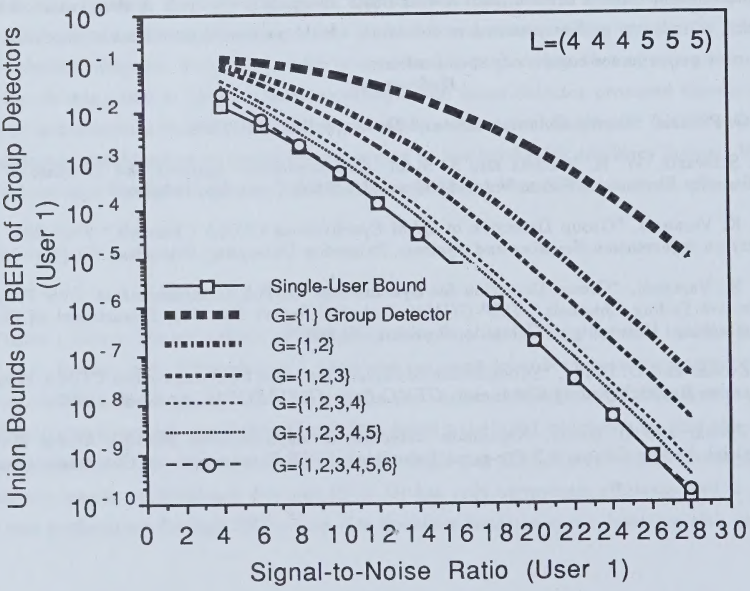
[Var92] M. K. Varanasi, "Group Detection in QAM Synchronous CDMA Channels," *Proc 26th Annual Conf on Information Sciences and Systems,* Princeton University, Princeton, NJ (Mar 1992).

[Var93b] M. K. Varanasi, "Group Detection for Synchronous CDMA Communication Over Frequency-Selective Fading Channels," *DSP-COMM Technical Report No. 514*, Department of Electrical Engineering, University of Colorado, Boulder, CO 80309.

[ZvB92] Z. Zvonar and D. Brady, "Suboptimum Multiuser Detector for Synchronous CDMA Frequensy-Selective Rayleigh Fading Channels," *CTMC Proc, GLOBECOM* , pp 82-86., 1992.

[ZvB93] Z. Zvonar and D. Brady, "Optimum Detection in Asynchronous Multiple-Access Multipath Rayleigh Fading Channels," *Pre-print*, Submitted, IEEE Transactions on Communications.


L=(4 4 4 5 5 5)
asymptotic efficiency (user 1)
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
1
2
3
4
5
6
group size


L=(4 4 4 5 5 5)
Union Bounds on BER of Group Detectors (User 1)
10^{0}
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
10^{-6}
10^{-7}
10^{-8}
10^{-9}
10^{-10}
Single-User Bound
G={1} Group Detector
G={1,2}
G={1,2,3}
G={1,2,3,4}
G={1,2,3,4,5}
G={1,2,3,4,5,6}
0
2
4
6
8
10
12
14
16
18
20
22
24
26
28
30
Signal-to-Noise Ratio (User 1)

The Highway PALM: A Stochastic Model to Capture Space and Time Dynamics in Wireless Communication Systems

William A. Massey and Ward Whitt
AT&T Bell Laboratories
Murray Hill, NJ 07974–0636

Abstract

In C. Palm's classic 1943 paper, "Intensity Variations in Telephone Traffic," he recognized the importance of queueing models with time-dependent arrival rates and proposed using the infinite-server $M_t/G/\infty$ model to study the corresponding $M_t/G/s/0$ loss model. We propose an extension of this approach to capture spatial dynamics as well as time-dependent arrival rates in wireless communication systems. For this purpose, we have developed the *Poisson-arrival-location-model* (PALM). In a PALM, arrivals generated by a nonhomogeneous Poisson process move independently through a general state space according to a location stochastic process. In this paper, we construct a special PALM to study communicating mobiles on a one-way single-lane semi-infinite highway. For the highway PALM considered here, key quantities are the call density, the handoff rate, the call-origination-rate density and the call-termination-rate density, which themselves are simply related by two fundamental conservation equations. We show that the highway PALM can be applied, together with independent superposition, to treat more complicated models.

1 Introduction

The rapidly evolving wireless communication systems present exciting challenges for teletraffic theory (see Lee [4]). Our purpose here is to present a new model which we believe can be used to help design and manage these wireless communication systems. The model we introduce here is a special case of the *Poisson-arrival-location model* (PALM) which we introduced in [6]. In a PALM a nonhomogeneous Poisson process generates arrivals that move independently through a general location state space according to some location stochastic process. A PALM enables us to model nonstationary behavior, non-Markovian behavior, and service dependence on space and time. The PALM is able to capture these important features because it does not represent interactions between customers. Our idea is that the PALM should be able to serve as a practical model for mobile phones, just as the infinite-server $M_t/G/\infty$ queue can serve as a practical model for telephone trunk groups. This idea for telephone trunk groups was advanced by Palm [10], which helps explain the name for our model.

Although the PALM ignores interactions between customers, it does capture many of the space-time dynamics of wireless communication systems. For a stationary telephone, only the time a call is in progress is important, so that service can be represented by the length of a telephone conversation. For mobile phones, both the location and time of a call in progress are important, so that service should be represented by a path through a location space as a function of time.

For moving automobiles, the two-dimensional space $\mathbb{R}^2$ is a natural location space, as in Viewgraph 1. For personal communication systems associated with people or airplanes, three-dimensional space $\mathbb{R}^3$ is a natural location space. When these spaces are partitioned into a finite set of cells, the resulting PALM corresponds to a standard finite-node, infinite-server network. This PALM can then model the number of phone conversations in each cell.

In this paper, our basic model will be for traffic on a one-way, single-lane, semi-infinite highway. This makes our location space equal to the interval $[0, \infty)$. The boundary point 0 will be called the *spatial origin*; it will mark the entrance point to the highway. A nonhomogeneous Poisson arrival process $\{ A(t) \mid -\infty < t < \infty \}$ counts the number of cars that enter the highway. In particular, $A(t)$ counts the number of arrivals up to time t, which we assume is finite with probability one. This nonhomogeneous Poisson process A is characterized by its arrival rate function α. To have $A(t)$ finite with probability one, we assume that $\int_{-\infty}^{t} \alpha(s)ds < \infty$ for all t. This paper is an abridged version of [7]; see [7] for further discussions.

It is significant that many formulas here (e.g., for means) do *not* depend on the Poisson assumption; they hold if A is an arbitrary point process (without multiple points) with time-dependent arrival-rate function α. The Poisson assumption for A is important for distribution conclusions (e.g., the Poisson property and stochastic independence).

In a general PALM, the movement of each customer after arrival can be characterized in terms of a stochastic process. Here, as a further simplification, we assume that the movement of the cars after arrival is described by a deterministic function $\chi(s, t)$ which gives the position at time t of a car entering the highway at time s. To specify the deterministic location function $\chi(s, t)$, we can draw on vehicular traffic theory, as in Gazis [1] and Haberman [2]. As often done there, we specify the function χ by a velocity field $v(x, t)$, which specifies the velocity at position x at time t. Our analysis of the highway PALM thus brings together teletraffic theory and vehicular traffic theory. Other recent efforts to do this are the papers by Meier-Hellstern, Alonso and O'Neill [8], Montenegro, Sengoku, Yamaguchi and Abe [9], and Seskar, Maric, Holtzman and Wasserman [11]. We anticipate much more of this in the future.

Even though we directly consider only one-way traffic on $[0, \infty)$ originating at 0, our analysis applies to much more general models, by virtue of superposition. To include other cars on the highway going the same direction with different starting points, and other cars going in the opposite direction, we can simply superimpose independent versions of the highway PALM model considered here. Moreover, general movement in $\mathbb{R}^2$ and $\mathbb{R}^3$ can be treated as superpositions of independent one-way traffic along paths in these spaces, as in Viewgraph 12. Hence the one-way highway is not only a natural model of interest in its own right, but it is a key building block for more complicated models.

It is important to keep in mind, though, that in all these models we are assuming that there are *no interactions* between different cars. The locations of all the cars are mutually independent, conditional on their arrival times. None of this precludes, however, the cars being influenced by their common space-time environment. For example, we cannot *directly* model a car deciding to slow down based on the actions of another car, but we can model this behavior *indirectly* by having all cars slow down in a specific region of the highway at a specific time.

Here is how this paper is organized. In Section 2, we construct the basic highway PALM and identify the four fundamental stochastic processes associated with it. They are $Q(x,t)$, the number of *calls in progress* in region $(0,x]$ before time t; $H(x,t)$, the number of *handoffs* at position x before time t; $C^+(x,t)$, the number of *call originations* in region $(0,x]$ before time t; and $C^-(x,t)$, the number of *call terminations* in region $(0,x]$ before time t. As in [6], they are Poisson processes when viewed properly; see Theorem 3.1. Moreover, their means (and thus their full distributions) are characterized by a call density $q(x,t)$, a handoff rate $h(x,t)$, a call-origination-rate density $c^+(x,t)$, and a call-termination-rate density $c^-(x,t)$, which are all deterministic functions that are related by two fundamental conservation equations, see (2.7) and (2.11).

In Section 3 we establish the main structural results for the four processes Q, H, C^+, and C^-. In Section 4 we analyze the traffic flow into and out of cells. No proofs are given here; all proofs are given in [7]. There we also describe how the highway PALM here can be applied to divide the highway into cells so that each cell can just handle its offered traffic. We contrast the case of fixed cell boundaries with dynamic cell boundaries (which are functions of time). Having dynamic cell boundaries is analogous to having dynamic (time-of-day) routing in a standard voice network. Fewer cells are required with dynamic cell boundaries. The highway PALM here provides a way to quantify the difference.

There we also indicate how we can describe the proportion of blocked originating calls in a cell and the proportion of blocked handoffs from one cell to another. For this purpose, we exploit MOL approximations; see Jagerman [3] and Massey and Whitt [5].

In [7] we also analyze a special case of a highway PALM in which all cars have constant velocity. For this special case, we can easily express the call density $q(x,t)$ in terms of the external arrival rate function α and other model features. We consider a simple numerical example to investigate the influence of the distributions beyond their means.

Finally, in [7] we also show how to exploit independent superposition in order to build more elaborate models. There we show that the two fundamental conservation equations still hold with appropriate definitions.

2 Highway PALM and Conservation Equations

For simplicity, we assume that each car goes through three successive phases: think mode, calling mode, and completion mode. Given that a car arrives to the highway at time s, the car (or its driver) is in *think mode* until some random *call origination time* T_s^+ when a call is placed. After the call is placed, the call is in progress and the car is in calling mode until some random *call termination time* T_s^-. After T_s^-, the call is finished and the car is in completion mode. Observing that we will always have $s \leq T_s^+ \leq T_s^-$, we will refer to $T_s^+ - s$ as the *think time* and $T_s^- - T_s^+$ as the *call holding time*. In the expanded version of this paper, we discuss extensions to the highway PALM in which cars can make multiple calls.

We now define the following quantities:

$$\begin{aligned}
\chi(s,t) &= \text{position of a car at time } t \text{ that enters the highway at time } s, \\
\tau(s,x) &= \text{time at position } x \text{ for a car entering the highway at time } s. \\
\sigma(x,t) &= \text{highway entrance time for a car to be in position } x \text{ at time } t, \\
v(x,t) &= \text{instantaneous velocity of a car in position } x \text{ at time } t.
\end{aligned}$$

Recall that we have stipulated that χ is a deterministic function, so that the same will be true for τ, σ and ν. To have $\sigma(x,t)$ and $v(x,t)$ well defined, we need to assume that two cars cannot be in the same place at the same time or, equivalently, that *cars cannot pass each other.* (In [7], we treat cases where cars can pass by using the principle of independent superposition.) In addition, we assume the following regularity conditions for χ:

1. Each arriving car starts at the spatial origin i.e., $\chi(s,s) = 0$.
2. All cars move forward continuously and tour the entire highway i.e.,
   (a) χ is strictly decreasing in the first argument and strictly increasing in the second argument.
   (b) $\chi(s,\cdot)$ is a continuous function that maps $[s,\infty)$ onto $[0,\infty)$ for all s.
3. The family of car trajectories have finite velocities and form a well-posed system i.e., χ is a continuously differentiable mapping from $\mathbb{R}^2$ to $\mathbb{R}$.

These assumptions enable us to show that we can construct τ, σ, and v from χ.

Let $Q(x,t)$ be the *number of cars with calls in progress* at time t within the interval of space $(0,x]$. We can express it in terms of Poisson integration as:

$$Q(x,t) \equiv \int_{\sigma(x,t)}^{t} 1_{\{T_s^+ \le t < T_s^-\}} dA(s). \tag{2.1}$$

where 1_B is the indicator function of the set B; i.e., $1_B(x) = 1$ if $x \in B$ and 0 otherwise. By "Poisson integration" we mean stochastic integration with respect to the nonhomogeneous Poisson process A. This integration can be simply defined as a finite sum for each sample path.

The process Q can also be constructed by applying Poisson integration to a *location process* $L_s(t)$, as we did in [6]. We define the *location space* to be the set of ordered pairs $(x,0)$ and $(x,1)$ for $x \in \mathbb{R}_+$, i.e., where x is a nonnegative real number. The pair $(x,0)$ corresponds to the car being at position x on the highway and in the think mode. The pair $(x,1)$ is defined by corresponds to the car being at position x in in calling mode. We can then let $L_s(t)$ be defined by

$$L_s(t) = \begin{cases} \Delta_* & s < t, \\ (\chi(s,t),0) & t \le s < T_s^+, \\ (\chi(s,t),1) & T_s^+ \le s < T_s^-, \\ \Delta^* & T_s^- \le s, \end{cases} \tag{2.2}$$

where Δ_* and Δ^* are the "pre-arrival" and "completion-mode" states as defined in [6]. This gives us an alternate construction of $Q(x,t)$ as

$$Q(x,t) = \int_{\sigma(x,t)}^{t} 1_{\{L_s(t)\in(0,x]\times\{1\}\}} dA(s). \tag{2.3}$$

We now assume in addition that σ is twice continuously differentiable, α is once differentiable, and for almost all s (with respect to the measure $\alpha(s)ds$), the distributions of both T_s^+ and T_s^- have densities. With these assumptions we can define a *call density* $q(x,t)$ where

$$q(x,t) \equiv \frac{\partial}{\partial x}\mathsf{E}[Q(x,t)]; \tag{2.4}$$

see (3.1) below. (The second derivative of σ is not needed here.)

We now suppose that the highway is partitioned into subintervals called *cells*. When a car is in a cell, its calls are handled by a designated *base station*. To describe the handoffs of calls from one cell to another at the cell boundary point x within the time interval $(-\infty,t]$, we use the function $\tau(s,x)$. Let one radio base station cover the cell or highway region $(w,x]$ and another cover $(x,y]$, where $0 \le w < x < y \le \infty$. Let $H(x,t)$ be the *number of handoffs* at x within the interval of time $(-\infty,t]$. We can express $H(x,t)$ in terms of Poisson integration as:

$$H(x,t) \equiv \int_{-\infty}^{\sigma(x,t)} 1_{\{T_s^+\le\tau(s,x)<T_s^-\}} dA(s). \tag{2.5}$$

The same conditions that allow us to define $q(x,t)$ also enable us to define a *handoff rate* $h(x,t)$ where

$$h(x,t) \equiv \frac{\partial}{\partial t}\mathsf{E}[H(x,t)]. \tag{2.6}$$

Paralleling the fundamental conservation equations of highway traffic, see p. 274 of Haberman [2], the call density q and handoff rate h are simply related, given the velocity (or equivalently the location function χ). The difference here is that we are talking about calls instead of cars. Here is our first fundamental conservation equation.

Theorem 2.1 *Given the velocity field v, the handoff rate h is related to the call density q by*

$$h(x,t) = v(x,t)q(x,t). \tag{2.7}$$

Finally, we let $C^+(x,t)$ and $C^-(x,t)$ be, respectively, the *number of calls that originate in the interval* $(0,x]$ *before time* t and *number of calls that terminate in the interval* $(0,x]$ *before time* t. Their Poisson integral representations are

$$C^+(x,t) \equiv \int_{-\infty}^{t} 1_{\{T_s^+\le\min(\tau(s,x),t)\}} dA(s) \tag{2.8}$$

and

$$C^-(x,t) \equiv \int_{-\infty}^{t} 1_{\{T_s^-\le\min(\tau(s,x),t)\}} dA(s). \tag{2.9}$$

Using the same conditions as for $q(x,t)$ and $h(x,t)$ (the second derivatives of σ are needed here), we define the *call-origination-rate density* $c^+(x,t)$ and the *call-termination-rate density* $c^-(x,t)$ to be

$$c^+(x,t) \equiv \frac{\partial^2}{\partial x \partial t}\mathsf{E}[C^+(x,t)] \quad \text{and} \quad c^-(x,t) \equiv \frac{\partial^2}{\partial x \partial t}\mathsf{E}[C^-(x,t)]. \tag{2.10}$$

It is significant that q, c^+ and c^- are also simply related, given the velocity. The relation is a partial differential equation that arises in semiconductor statistics; see p. 51 of Sze [12]. This is our second fundamental conservation equation

Theorem 2.2 *Given the velocity field v, call-origination-rate density c^+ and call-termination-rate density c^-, the call density q solves the partial differential equation:*

$$\frac{\partial}{\partial t}q(x,t) + \frac{\partial}{\partial x}v(x,t)q(x,t) = c^+(x,t) - c^-(x,t). \tag{2.11}$$

It can be shown that any formula from this highway PALM model that only uses the first moments of the random processes constructed here will still be true when the arrival process $\{ A(t) \mid -\infty < t < \infty \}$ is any simple point process. Hence the key Theorems 2.1 and 2.2 hold in greater generality.

3 The Fundamental Results

We now describe in more detail the four fundamental processes Q, H, C^+, and C^- defined in Section 2. The following can be regarded as a consequence of Theorem 2.1 of [6].

Theorem 3.1 *The following results hold:*

(a) For all real t, $\{ Q(x,t) \mid x \geq 0 \}$ is a Poisson process with

$$\mathsf{E}[Q(x,t)] = \int_{\sigma(x,t)}^{t} \mathsf{P}\left(T_s^+ \leq t < T_s^-\right)\alpha(s)ds. \tag{3.1}$$

(b) For all non-negative x, $\{ H(x,t) \mid -\infty < t < \infty \}$ is a Poisson process with

$$\mathsf{E}[H(x,t)] = \int_{-\infty}^{\sigma(x,t)} \mathsf{P}\left(T_s^+ \leq \tau(s,x) < T_s^-\right)\alpha(s)ds. \tag{3.2}$$

(c) Both $\{ C^+(x,t) \mid x \geq 0, -\infty < t < \infty \}$ and $\{ C^-(x,t) \mid x \geq 0, -\infty < t < \infty \}$ are two dimensional Poisson processes with

$$\mathsf{E}[C^+(x,t)] = \int_{-\infty}^{\sigma(x,t)} \mathsf{P}\left(T_s^+ \leq \tau(s,x)\right)\alpha(s)ds + \int_{\sigma(x,t)}^{t} \mathsf{P}\left(T_s^+ \leq t\right)\alpha(s)ds. \tag{3.3}$$

and

$$\mathsf{E}[C^-(x,t)] = \int_{-\infty}^{\sigma(x,t)} \mathsf{P}\left(T_s^- \leq \tau(s,x)\right)\alpha(s)ds + \int_{\sigma(x,t)}^{t} \mathsf{P}\left(T_s^- \leq t\right)\alpha(s)ds. \tag{3.4}$$

As in Remark 2.3 of [6], it is significant that the mean formulas in (3.1) and (3.2) do *not* depend on the arrival process A being a Poisson process. Of course, the Poisson process conclusions in Theorem 3.1 *do* depend on A being Poisson. As in [6], let $\oplus$ denote the sum of two random quantities, where the summands are stochastically independent.

Theorem 3.2 *The following independence relations hold for the random processes Q, H, C^+, and C^-:*

(a) For all space-time pairs (x,t) and (x',t') with $\sigma(x,t) \geq \sigma(x',t')$, $Q(x,t)$ is independent of $H(x',t')$.

(b) The process C^- is independent of both processes Q and H.

(c) Finally, for all space-time pairs (x,t),

$$C^+(x,t) = Q(x,t) \oplus H(x,t) \oplus C^-(x,t). \tag{3.5}$$

4 Fundamental Theorem for Cell Traffic

At this point, we want to think more in terms of cells, so let $(x,y]$ be a generic cell where of course $x < y$. Note that calls can arrive at a cell in two ways: (1) by a car already in the cell initiating a call, and (2) by a handoff of a call in progress for a car entering the cell. Similarly, calls can leave a cell in two ways: (1) by a car already in the cell terminating a call, and (2) by a handoff of a call in progress for a car leaving the cell. In this section we describe these four processes and their relationships.

If we only focus on whether or not a call is in progress in a cell and suppress the position within a cell, then a partition of the highway into a finite number of cells allows us to represent the highway PALM as an $(M_t/G_t/\infty)^N/G_t$ network in the terminology of [6]. Then cells are nodes and N is the number of nodes. Each cell has an external arrival process corresponding to call originations in the cell and a departure process corresponding to call terminations in the cell. There are also flows from node $i-1$ to i for $2 \leq i \leq N+1$ corresponding to handoffs of calls in progress when a car passes a cell boundary.

We now focus on one cell within this $(M_t/G_t/\infty)^N/G_t$ network. From [6], we know that this cell itself can be regarded as an $M_t/G_t/\infty$ queue. We do a direct analysis here. The results below can be extended to more general location spaces. However, here we exploit the fact that no call can re-enter a cell after it has left; e.g., see Theorem 3.5 of [6].

We now define $C^+_{(x,y]}(t)$ and $C^-_{(x,y]}(t)$ to be respectively the *number of calls that originate and terminate in cell* $(x,y]$. We can express them in terms of C^+ and C^- as:

$$C^+_{(x,y]}(t) \equiv C^+(y,t) - C^+(x,t) \quad \text{and} \quad C^-_{(x,y]}(t) \equiv C^-(y,t) - C^-(x,t). \tag{4.1}$$

Similarly, if we define $Q_{(x,y]}(t)$ to equal the number of calls in cell $(x,y]$ at time t, then

$$Q_{(x,y]}(t) \equiv Q(y,t) - Q(x,t). \tag{4.2}$$

By Theorem 3.1, $Q_{(x,y]}(t)$ has a Poisson distribution with mean $q_t(x,y) = \int_x^y q(z,t)dz$.

As in Section 3, let $\oplus$ denote the sum of two random variables that are stochastically independent. We write $X = Y \ominus Z$ if $X \oplus Z = Y$.

Theorem 4.1 *For any given cell $(x,y]$, the following results hold:*

(a) The arrival process of calls to cell $(x,y]$ is Poisson and equals

$$\left\{ H(x,t) \oplus C^{+}_{(x,y]}(t) \mid -\infty < t < \infty \right\}. \tag{4.3}$$

(b) The departure process of calls from cell $(x,y]$ is Poisson and equals

$$\left\{ H(y,t) \oplus C^{-}_{(x,y]}(t) \mid -\infty < t < \infty \right\}. \tag{4.4}$$

(c) Finally, we have for all t,

$$Q_{(x,y]}(t) = [H(x,t) \oplus C^{+}_{(x,y]}(t)] \ominus [H(y,t) \oplus C^{-}_{(x,y]}(t)]. \tag{4.5}$$

References

[1] Gazis, D.C., *Traffic Science*, Wiley, New York, 1974.

[2] Haberman, R., *Mathematical Models: Mechanical Vibrations, Population Dynamics, and Traffic Flow*, Prentice-Hall, Inc., New Jersey, 1977.

[3] Jagerman, D. L., Nonstationary Blocking in Telephone Traffic, *Bell System Tech. J.* 54 (1975) 625–661.

[4] Lee, W.C.Y., *Mobile Cellular Telecommunications Systems*, McGraw-Hill, New York, 1989.

[5] Massey, W.A. and Whitt, W., On the Modified-Offered-Load Approximation for the Nonstationary Erlang Loss Model, *Annals of Applied Probability* (to appear).

[6] Massey, W.A. and Whitt, W., Networks of Infinite Server Queues with Nonstationary Poisson Input. *Queueing Systems*, 13 (1993), 183–250.

[7] Massey, W.A. and Whitt, W., A Stochastic Model to Capture Space and Time Dynamics in Wireless Communications Systems, submitted to *Annals of Applied Probability.*

[8] Meier-Hellstern, K.S., Alonso, E. and O'Neill, D. R., The Use of SS7 and GSM to Support High Density Personal Communication Systems, in *Third WINLAB Workshop on Third Generation Wireless Networks*, April 1992.

[9] Montenegro, G., Sengoku, M., Yamaguchi, Y. and Abe, T., Time-Dependent Analysis of Mobile Communication Traffic in a Ring-Shaped Service Area with Nonuniform Vehicle Distribution. *IEEE Trans. Vehicular Tech.* 41 (1992) 243–254.

[10] Palm, C., Intensity Variations in Telephone Traffic, *Ericsson Technics* 44 (1943) 1–189 (in German). (English translation by North-Holland, Amsterdam, 1988).

[11] Seskar, I., Maric, S. V., Holtzman, J. and Wasserman, J., Rate of Location Area Updates in Cellular Systems, in *Third WINLAB Workshop on Third Generation Wireless Networks*, April 1992.

[12] Sze, S.M., *Physics of Semiconductor Devices* (Second Edition), John Wiley & Sons (1981).

On the Capacity Bounds in Cellular Systems

Z. J. Haas, J. H. Winters, and D. S. Johnson
AT&T Bell Labs, Room 4F-501, P. O. Box 3030, Holmdel, NJ 07733

ABSTRACT In this paper, we investigate the capacity of cellular systems. In particular, we study how the reuse factor can be improved given the knowledge of the mobiles' locations; i.e., we evaluate the minimum number of channels required to support a cellular infrastructure with a given number of mobiles in each cell. We assume that the mobiles' locations are sampled from the uniform random distribution or are fixed on a uniform grid. Moreover, we show the effect of a number of parameters, such as the number of mobiles per cell, the minimum allowable signal-to-interference ratio, and limited knowledge of mobile location. The assumption of a single interferer, used in our study, is also justified.

1 Introduction

A large body of research has been published on the performance of cellular systems (e.g., [1] - [5]). Most of these papers present and analyze schemes in which no knowledge about the mobiles' locations, besides the mobiles' associated base-stations, is taken into the account. Thus, a channel is allocated to a cell, rather than a mobile. In this work, we assume that there is a mechanism by which mobiles can measure the amount of interference that they receive on different channels. Thus, in our work, the allocation of channels is to mobiles rather than cells. We show the amount of improvement that such a scheme carries.

In fixed channel allocation (FCA), a channel assigned to a cell can be reused according to some (fixed) cell reuse pattern ([1]). For example for cell reuse pattern of 7, a channel used in the cell A is not reused at any other cell whose distance to the cell's A base-station is closer than about three cell "radii." Similarly, for cell reuse pattern of 3, the corresponding distance is 2 "radii."

Because, at any time, some cells may require more capacity than others, the FCA suffers from inefficient use of the radio spectrum. To correct this limitation of FCA, Dynamic Channel Allocation (DCA) schemes have been proposed.[1] In fact, a multiplicity of DCA schemes have been investigated by various researchers, showing the advantage of one scheme over another under some traffic and other assumptions ([5, 7, 8, 9, 10, 11]). The traffic adaptation based DCA schemes rely on the idea that, instead of dedicating the channels to cells, the channels are "placed" in a pool, and are allocated on demand to the cells based on some allocation rules.

In both FCA and the traffic adaptation DCA, the channels are usually allocated to cells based on the assumption that a mobile may be located anywhere within the boundary of that cell. To reuse the channel in another cell, that cell must be a given distance away. Thus, for both the schemes, the "packing"[2] of channels is not maximal. These schemes suffer from the fact that the fixed reusability factor may be too pessimistic; e.g., mobiles may not interfere with each other even if they're in adjacent cells and the reuse pattern corresponds to distance of 2 radii. In the interference adaptation DCA schemes mobiles measure the amount of interference to determine the usability of a channel. Such algorithms can achieve maximum packing. An example of a system based on this principle is the DECT standard ([12]).

The maximum packing corresponds to the minimum number of channels required to support a given amount of traffic in each cell. Let's clarify this: a maximum packing for the specific mobile distribution means that whatever the allocation, the computed number of channels is the minimum number of channels that must be allocated to provide connection to all the mobiles; i.e., the number of channels with maximum packing is a lower bound on the number of channels required in any cellular system.

---

[1] We distinguish here between DCA schemes that respond to traffic changes (traffic adaptation) and to changes in the level of measured interference (interference adaptation). For example, see [6].

[2] The channel "packing" refers to the areas where a channel cannot be reused and how closely these areas are spaced.

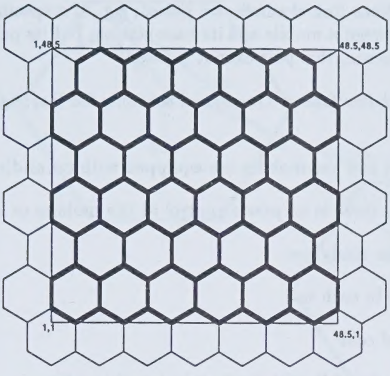


Figure 1: Cells structure

The minimum number of channels under the maximum packing condition is a static measure. In other words, when the mobiles move, the particular assignment may not be valid anymore and may need to be recomputed. This may lead to reassignment of the channels of many (if not all) of the mobiles in the system; i.e., the number of channel reassignments required may be prohibitively large. Thus the maximum packing strategy serves for theoretical comparison purposes only.

In general, the bound will vary with the location of the mobiles. However, our results show that with randomly-located mobiles (the random locations drawn from the uniform distribution), the variation in the bound for different realizations of the random locations is relatively small. Therefore, we use Monte Carlo simulation to determine the bound for a few realizations and use these results to obtain an approximate bound for most cases of randomly located users (i.e., small probability of blocking). The reader is referred to [13] for the discussion of blocking probabilities of DCA schemes.

In practice, we may not need to assign channels to all mobiles, but can block some small percentage of arriving calls. Here we ignore this fact, which would result in somewhat fewer channels being required.

Also, because of the time-variations in the traffic, the required number of channels will also vary. Using the results presented here and knowing the traffic variations (or distribution), one can readily determine the distribution of the required number of channels with the traffic variations.

This paper is organized as follows: In the next section, we describe the model of the cellular network and list the major assumptions used throughout this work. In section 3, we show the effect of the different parameters on the results of the maximum packing strategy. For example, we consider the effect of channel allocation based on the distance of the mobile from the base station, without knowledge of the actual location of the mobile. Finally, in section 4, we summarize and conclude the study.

2 Assumptions, Parameters, and Methodology

The following model is used in this work:

1. The cellular structure is composed of 33 internal cells, as shown in Figure 1, surrounded by additional cells. The purpose of the additional cells is to eliminate the "boundary effects," which would lead to too optimistic results (too small number of required channels). In other words, the number of channels required for this structure should be approximately the same as the number required for an arbitrary large cellular structure. A base-station is located in the middle of each cell, and serves the mobiles that are located within the boundaries of the cell.

2. The *single interferer* assumption: In our study we assume that there is only one interfering link (i.e., with the same assigned channel) to a communication link under study. In reality, there may be several other links communicating on the same channel and contributing interference to the link in question. However, as shown in section 3.6, when a single interferer is used to calculate channel allocation, there is only a small probability that the resulting total interference of all the links exceeds the designed-for interference level. Thus, the *single interferer* assumption is justified.

3. The up link and the down link channels are paired; e.g., if a specific up link channel is available for communication between a mobile and its base-station, but its paired down link channel suffers from too much interference, this pair cannot be used.

4. We do not consider *shadow fading.* Our model accounts for Rayleigh fading and propagation loss only.

5. Both the base-stations and the mobiles are equipped with omni-directional antennas.

6. Except for section 3.5, there is no power control at the mobiles or at the base stations.

The parameters used in the study are:

- n - number of mobiles in each cell
- c - the total number of cells
- N - the total number of mobiles ($= n \cdot c$)
- α - the *interference radius* - the required ratio between the distance from the base-station to the interferer and the distance from the base-station to the mobile
- P - the power transmitted by the mobile or the base-station when no power control is performed
- S - the signal's received power
- I - the level of received interfering signal
- r - the power of radio signal loss (r=2 for free space propagation without the multipath effect; for mobile radio r=3.8 is usually assumed)
- N_c - number of channels required to accommodate the n users under given conditions
- x - the distance of a mobile to its base-station
- d - the distance of a mobile to another base-station (other than its base-station)

The study evaluates the minimum number of channels (referred to here also as colors) assigned to the mobiles under given operating conditions, such that given interference conditions are satisfied. The operating conditions refer to the knowledge or lack of knowledge of the location of the mobiles. The interference conditions refer to the acceptable level of interference, so that two mobiles can be assigned the same channel.

In this work, we consider FDMA assignment. Thus, two TDMA slots are considered as separate channels that do not suffer from cross interference. In this sense, the results have also some limited applicability to the CDMA case.

The number of channels is evaluated by constructing a matrix (referred to here as *compatibility matrix* of dimension $N \times N$); i.e., each mobile in the whole system is evaluated with each other mobile to determine whether the two mobiles can be assigned the same channel (i.e., whether they are "compatible") based on the interference conditions. A graph is then composed, where each mobile in the system corresponds to a vertex in the graph. Two vertices are interconnected by an edge in the graph, if and only if the two mobiles represented by the two vertices are "incompatible"; i.e., they cannot be assigned the same channel. A set of graph coloring algorithms is then employed to find the minimum number of colors to color the vertices in the graph, such that no two vertices interconnected by an edge have the same color. Each color corresponds then to a channel assigned to the mobiles that the vertices colored in that color represent. Thus, the number of colors required equals the number of channels. (This problem can also be posed as finding the minimum number of cliques[3] that cover all the vertices in the complementary graph.[4] A clique in the complementary graph, then, corresponds to a channel.) For a small number of vertices, finding the number of colors may not be a considerable obstacle. However,

---

[3] A *clique* is a subgraph in which every two vertices are interconnected by an edge.

[4] The complementary graph is a graph that has edge joining two nodes if and only if they were not so joined in the original graph.

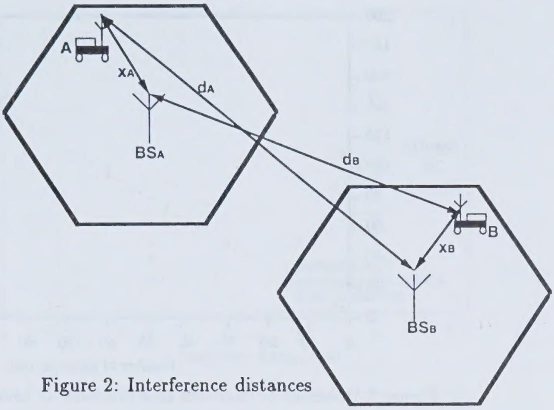


Figure 2: Interference distances

to obtain any meaningful statistics, instances with large number of mobiles need to be evaluated. Thus, since the graph coloring problem is an NP-complete problem ([14]), finding the number of colors can be a complex task, as is the problem of finding the minimum number of cliques. In this work, we thus use an algorithm suggested by [15] that, while not finding the exact minimum number of colors required, is remarkably effective at finding good upper bounds on that number for the types of graphs we create. We then complement our upper bounds with lower bounds determined by heuristics for the maximum clique problem, thus pinning the precise value within a narrow range, and indeed often determining it exactly.

Other approaches to cope with the large complexity of the problem have been proposed. In particular, simulated annealing and neural networks were used ([16, 17]).Our algorithm is significantly faster than the simulated annealing and neural net approaches and finds bounds that are typically just as good, or, in the latter case significantly better.

Such schemes usually suffer from the uncertainty whether the obtained solution is, in fact, optimal and what is the error from the optimal solutions. Our "brute force" scheme presented here provides either an 'exact' solution or tight bounds within which the optimal result exists.

In [6], FCA was compared with interference adaptation DCA through analytical methods to obtain lower and upper bounds on the performance of interference adaptation DCA. However, the bounds presented there are very loose. As mentioned before, our bounds are much tighter, usually differing by only a few percent.

2.1 The Interference Conditions

In general, the signal-to-interference (SIR) ratio determines the ability of the receiver to recover the signal of power S in the presence of interference of power I. We assume that there is some minimum SIR, termed SIR_{min}, required for sufficient system operation. Thus:

$$\mathrm{SIR} = \frac{S}{I} = \frac{\frac{P}{x^r}}{\frac{P}{d^r}} = \left(\frac{d}{x}\right)^r \geq \mathrm{SIR}_{min}. \tag{1}$$

We define α to be:

$$\alpha = \sqrt[r]{\mathrm{SIR}_{min}}. \tag{2}$$

Thus to achieve sufficient performance (i.e., SIR$\geq$ SIR_{min}):

$$\frac{d}{x} \geq \alpha. \tag{3}$$

Assume there are two mobiles, one in each cell, as shown in Figure 2. If the two mobiles are assigned the same up link channel and the two base-stations are assigned the same down link channel, then the SIR of the base-station$_A$ to mobile A communication, resulting from interfering communication between base-station$_B$ to mobile B is:

$$\mathrm{SIR}_A = \frac{\frac{P}{{x_A}^r}}{\frac{P}{{d_A}^r}} = \left(\frac{d_A}{x_A}\right)^r \tag{4}$$

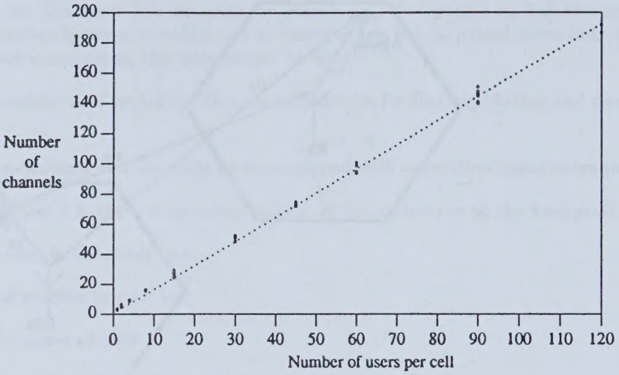


Figure 3: Number of channels as a function of mobile density

and the SIR of mobile A to base-station$_A$ communication, resulting from interfering communication between mobile B and base-station$_B$ is:

$$\mathrm{SIR}_{BS_A} = \frac{\frac{P}{x_A{}^r}}{\frac{P}{d_B{}^r}} = (\frac{d_B}{x_A})^r \tag{5}$$

Similarly,

$$\mathrm{SIR}_{B} = \frac{\frac{P}{x_B{}^r}}{\frac{P}{d_B{}^r}} = (\frac{d_B}{x_B})^r \tag{6}$$

$$\mathrm{SIR}_{BS_B} = \frac{\frac{P}{x_B{}^r}}{\frac{P}{d_A{}^r}} = (\frac{d_A}{x_B})^r \tag{7}$$

Since each one of the SIR's must satisfy the condition that SIR$\geq$ SIR$_{min}$, equations (4) – (7) correspond to the following set of conditions:

$$d_A \geq \alpha \cdot x_A \qquad d_B \geq \alpha \cdot x_A \qquad d_B \geq \alpha \cdot x_B \qquad d_A \geq \alpha \cdot x_B \tag{8}$$

There are n mobiles per cell, randomly located throughout the cell. The random location is drawn from the uniform distribution. The conditions (8) are checked for every pair of mobiles and two mobiles are declared compatible if the conditions (8) are satisfied. Otherwise, the mobiles are incompatible. The results are arranged in the compatibility matrix, which is used as an input to our graph-coloring and clique-finding algorithms. From these we obtain upper and lower bounds (typically quite tight) on the minimum number of channels required to support communication to/from all the mobiles with SIR greater than SIR$_{min}$.

While knowing the location of each mobile may not be possible, all that is necessary for maximum packing is the knowledge of the signal levels of each mobile's signal at all the base stations and the knowledge of each base station's signal at all the mobiles. With such knowledge, the signal-to-interference ratio for an allocation can be determined.

3 The Bound on the Number of Channels

In this section, we show the effect of the various parameters in our system on the minimum number of channels, N_c.

3.1 The Effect of the Number of Mobiles per Cell

The number of required channels, N_c, as a function of mobile density is shown in Figure 3. Four runs were performed for the following number of mobiles per cell: n = 1, 2, 4, 8, 15, 30, 45, 60, 90, 120. As demonstrated in this figure, the number of channels increases linearly with the mobile density, measured

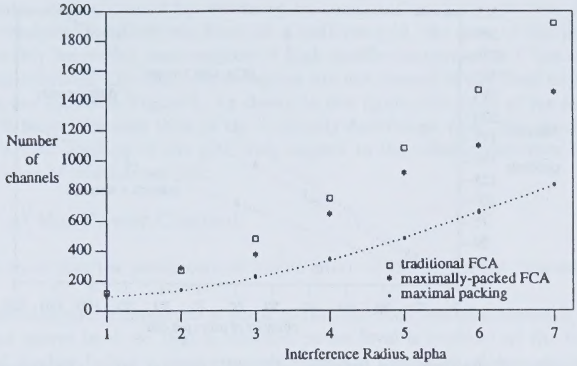


Figure 4: Number of channels as a function of the interference radius

here in mobiles-per-cell. The results shown in this figure were obtained with $\alpha = 2.0$, which corresponds to a reuse pattern of 3. The slope of the curve in Figure 3 is about 1.6. Thus, we've shown that for $\alpha = 2$, the reduction in the number of channels offered by maximum packing vs. fixed reuse pattern is approximately a factor of 2; i.e., maximum packing requires only about half as many channels as FCA. Moreover, because of the linear behavior, many of our results that follow are independent of the actual number of mobiles per cell. Note that, because the number of mobiles per cell is constant in our model, traffic adaptation DCA has no advantage over FCA.

3.2 The Effect of the Interference Radius

The curve in Figure 4 shows the dependence of the number of required channels on the interference radius, α, for the maximum packing scheme. The results were obtained for 90 mobiles per cell. This figure also shows the relative reduction of the number of channels with the maximum packing scheme, compared with the following maximally-packed FCA scheme. The maximally-packed FCA scheme assumes that the channel assignment is fixed to the cells, but is not necessarily repetitious. This is in contrast with the "traditional" FCA schemes, where the assignments to the cells are replicated and appear as a repetitious pattern. The numbers for the maximally-packed FCA scheme were obtained by running our maximum packing algorithm with a single mobile per cell located on the "circumference" of the cell, at the point that is closest to the circumference of the cell of the interfering mobile (i.e., the distance is the shortest distance between two points, one point on the circumference of each cell). For a given interference radius, the minimum number of colors equals the re-use factor.

For comparison purposes, the traditional FCA scheme is also shown in Figure 4. The number of channels for the traditional scheme were obtained by the following formula (taken from [5] with adjustments to reflect our definitions):

$$N_c(\text{FCA}) = \frac{n}{3}(\alpha + 1)^2. \tag{9}$$

As clearly shown in the figure, the actual gain achieved from the maximum packing algorithm increases with the interference radius as compared with the traditional FCA and, compared to the maximally-packed FCA, maximum packing achieves an approximately constant gain of 40% over the range of $\alpha = 2$ to $\alpha = 7$. For example, for interference radii of 2 and 3.6, the corresponding fixed reuse patterns for the traditional FCA are 3 and 7, respectively. The number of channels with the maximum packing is approximately 145 and 230, respectively, and the reduction in the number of channels is, thus, 52% and 67%, respectively. For $\alpha = 2$ and $\alpha = 4$, the maximally-packed FCA requires 270 and 630 channels, respectively, while the maximum packing scheme requires 145 and 350 channels, respectively, reducing the number of channels by 46% and 44%, respectively.

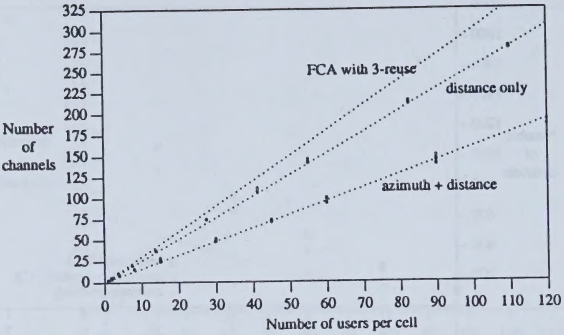


Figure 5: Number of channels as a function of mobile density, when the azimuth is not known

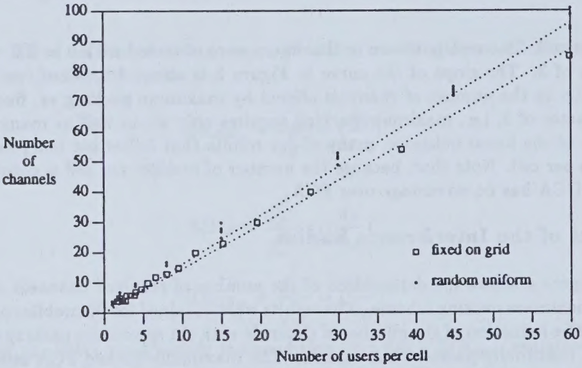


Figure 6: Number of channels as a function of mobile density for uniformly distributed mobiles

3.3 The Effect of No Knowledge of Direction (Azimuth)

By measuring the received power,[5] the base-station may be able to determine the distance to the mobile, without knowing its actual position; i.e., ignorance of the azimuth. When the maximum packing is performed without the knowledge of the azimuth, the gain of the maximum packing is reduced, as shown in Figure 5. Curves "azimuth+distance" and "distance only" represent the cases when the actual location of the mobile is known, and when only the distance of the mobile from the base-station is known, respectively. For comparison purposes, the curve "FCA with 3-reuse" shows the results for FCA with 3-reuse pattern. Results in Figure 5 were obtained with $\alpha = 2.0$. Thus, for example, for 90 mobiles per cell and for $\alpha = 2.0$, the number of required channels with no knowledge of direction is 229, or a reduction of 24%. This is about half of the reduction of 52%, when the direction is known.

When the power levels, rather than the actual mobiles' locations are used (and assuming no shadow fading), no knowledge of the direction is equivalent to the situation where the information about the received power levels is not shared among the base-stations (and among mobiles); i.e., each base-station knows the signal levels of its mobiles only. Similarly, every mobile knows the signal levels from all the base stations, but not other mobiles' levels.

3.4 The Effect of Mobile Distribution within the Cell

---

[5] When power control is performed, additional information on the transmitted power needs to be conveyed to the receiver site.

The previous results were obtained by assuming random (uniform) distribution of mobiles within the cell. When the mobiles' locations are fixed on a uniform grid, the gain of the maximum packing algorithm is considerably improved, since regions of high mobile concentration ("hot spots") can occur under the random distribution. Obviously, such regions are not present in the fixed uniform distribution case, and the results are shown in Figure 6. As shown in this figure, the slope of the curve is about 1.4, indicating about 15% fewer channels than in the randomly distributed case. The results were obtained with $\alpha = 2.0$. The actual position of the grid with respect to the cellular structure has no noticeable effect with large number of mobiles per cell.

3.5 The Effect of the Power Control

In this section we show that the power control has no effect on the number of channels, as investigated in this study.

In the power controlled environment, as is most commonly used, mobiles transmit their signal with dynamically adjusted power level, so that a constant power level is received at the base stations; i.e., the effect of loss and shadow fading is compensated. Similarly, the power of base stations' transmission is adjusted so that the signal received at the mobiles is of constant level.

Assume that the power received by the mobile and the base-station is constant and labeled P_c. The actual power transmitted from/to a mobile located at a distance x_m is thus $P_c \cdot (x_m)^r$. Consequently, the equations $(4-7)$ can be rewritten as:

$$\mathrm{SIR}_A = \frac{\frac{P_c \cdot x_A{}^r}{x_A{}^r}}{\frac{P_c \cdot x_B{}^r}{d_A{}^r}} = (\frac{d_A}{x_B})^r \tag{10}$$

$$\mathrm{SIR}_{BS_A} = \frac{\frac{P_c \cdot x_A{}^r}{x_A{}^r}}{\frac{P_c \cdot x_B{}^r}{d_B{}^r}} = (\frac{d_B}{x_B})^r \tag{11}$$

$$\mathrm{SIR}_B = \frac{\frac{P_c \cdot x_B{}^r}{x_B{}^r}}{\frac{P_c \cdot x_A{}^r}{d_B{}^r}} = (\frac{d_B}{x_A})^r \tag{12}$$

$$\mathrm{SIR}_{BS_B} = \frac{\frac{P_c \cdot x_B{}^r}{x_B{}^r}}{\frac{P_c \cdot x_A{}^r}{d_A{}^r}} = (\frac{d_A}{x_A})^r \tag{13}$$

Thus the four conditions that the locations of the two mobiles need to fulfill to be assigned the same channel (i.e., to be compatible) are:

$$d_A \geq \alpha \cdot x_B \qquad d_B \geq \alpha \cdot x_B \qquad d_B \geq \alpha \cdot x_A \qquad d_A \geq \alpha \cdot x_A, \tag{14}$$

which are exactly the conditions (8), in different order. Thus we conclude that two mobiles' locations are compatible, regardless of whether power control is performed; i.e., our results are applicable to both the power-controlled and the non-controlled cases. A similar conclusion can be drawn from [18]. Note, that when the power loss is not fully compensated for (as in [19]) our conclusions do not apply anymore.

3.6 The Effect of the "Single-Interferer" Assumption

In this study, we've assumed a single interferer. This assumption implies that in each case, there is one interferer that is significantly closer than all the other interferers. In this section, we justify the single interferer assumption.

In Figure 7, the probability distribution function of the mobiles' SIR is shown for the cases of $r = 3.0$, 3.5, and 4.0. The probability distribution was obtained for $\alpha = 2.0$ by analyzing the actual assignments generated by the graph-coloring algorithms. For $\alpha = 2.0$, the corresponding SIR_{min}'s for the above three values of r are: 9 dB, 10.5 dB, and 12 dB, respectively. Vertical lines, corresponding to these values are shown in the Figure 7. Considering all the interferers, Figure 7 shows that, when the single interferer assumption is made, the percentages of mobiles with SIR worse than SIR_{min} are: 20%, 10%, and 5%, for $r = 3.0, 3.5, 4.0$, respectively. We conclude that the results generated using the single interferer assumption contain channel assignments with relatively small percentage of mobiles violating SIR_{min}. Consequently, the error in the minimum number of channels due to the single interferer assumption for $r \geq 3.5$ can be considered negligible.

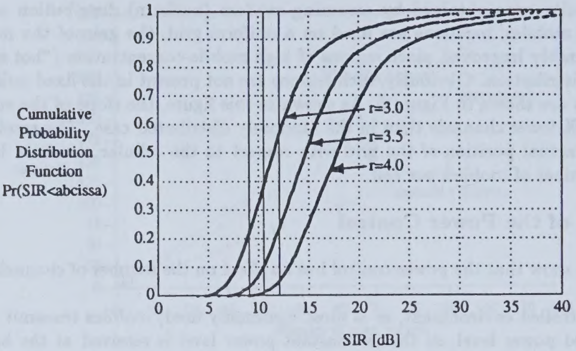


Figure 7: Cumulative probability distribution function of the mobiles' SIR

4 Concluding Remarks

In this paper, we have investigated the performance of the maximum packing algorithm for cellular structures. Our results are innovative in the sense that the channel assignments are done based on the knowledge (or limited knowledge) of the mobiles' locations. Thus, the cellular structure does not, by itself, limit the reuse of channels. An example of such a system is one in which the mobiles (and the base stations) measure the amount of interference to determine the available channels. The assignments are then done based on these measurements.

Our investigation used heuristics for graph-coloring and clique-finding to obtain tight bounds on the minimum number of channels needed in each situation. The heuristics were chosen both for their speed and for the tightness of bounds they yielded, and in these respects seemed to significantly outperform other methods such as simulated annealing and neural nets.

The results of this study indicate that the maximum packing scheme presented here can reduce the number of channels by nearly a factor of 2, for an interference radius of 2.0. More specifically, approximately 40% fewer channels are required in the maximum packing scheme than in maximally-packed (i.e., not necessarily repetitious) FCA, independent of the interference radius. Furthermore, the improvement with respect to the (traditional) FCA scheme is even larger and increases with the interference radius.

When only the distance (or power level) between each mobile and its nearest base station is known, the reduction in the number of channels is somewhat modest, about 25% for interference radius of 2.0. More uniform mobile distribution reduces the number of required channels, since mobile clustering tends to reduce the channel reuse. We found about a 15% difference in the number of channels between the random (uniform) and fixed (on square grid) cases (for interference radius of 2.0).

Finally, we have demonstrated that power control that compensates for power loss has no effect on the number of channels and that the assumption of a single interferer, as used in our study, was justified, showing that only about 10% of mobiles experience worse than designed for signal-to-interference ratio as the result of the single interferer assumption (assuming $r=3.5$).

References

[1] V. H. MacDonald, "Advanced Mobile Phone System: The Cellular Concept," *Bell Syst. Tech. Journal.*, vol. 58, no. 1, pp. 15-41, Jan. 1979.

[2] S. Nanda and D. J. Goodman, "Dynamic Resources Acquisition: Distributed Carrier Allocation for TDMA Cellular Systems," in *Third Generation Wireless Information Networks,* S. Nanda and D. J. Goodman, editors, Kluwer Academic Publishers, 1992.

[3] V. Prabhu and S. S. Rappaport, "Approximate Analysis of Dynamic Channel Assignment in Large Systems with Cellular Structure," *IEEE Transactions on Communications,* vol. COM-22, pp. 1715-1720, October 1974.

[4] J. C. -I. Chuang, "Performance Issues and Algorithms for Dynamic Channel Assignment," *IEEE Journal on Selected Areas in Communications: Part I,* vol. 11, no. 6, August 1993.

[5] W. C. Jakes, Jr, (editor), *Microwave Mobile Communications,* John Wiley and Sons, 1974.

[6] J. Zander and H. Eriksson, "Asymptotic Bounds on the Performance of a Class of Dynamic Channel Assignment Algorithms," *IEEE Journal on Selected Areas in Communications,* vol. 11, no. 6, August 1993, pp. 926-933.

[7] T. J. Kahava and N. D. Georganas, "A Hybrid Channel Assignment Scheme in Large-Scale, Cellular-Structured Mobile Communication Systems," *IEEE Transactions on Communications,* vol. COM-26, pp. 432-438, April 1978.

[8] S. M. Elnoubi, R. Singh, and S. C. Gupta, "A New Frequency Channel Assignment Algorithm in High Capacity Mobile Communication Systems," *IEEE Transactions on Vehicular Technology,* vol. VT-32, pp. 125-131, August 1982.

[9] D. E. Everitt and N. W. Macfadyen, "Analysis of Multicellular Mobile Radiotelephone Systems with Loss," *British Telecommun. Technol. Journal,* vol. 1, no. 2, pp. 37-45, Oct. 1983.

[10] M. Sengoku, M. Kurata, and Y. Kajitani, "Rearrangements of Switching Networks and Its Application to a Mobile Radio Communication System," in *IEEE Int. Conf. Circuits Comput.*, 1980, pp. 402-405.

[11] P-A. Raymond, "Performance Analysis of Cellular Networks," *IEEE Transactions on Communications,* vol. 39, no. 12, Dec. 1991.

[12] "DECT Services and Facilities Requirements Specification," RES3S (3) 90, Rev 5, 4th May 1990.

[13] J. F. Whitehead, "Performance and Capacity of Distributed Dynamic Channel Assignment and Power Control in Shadow Fading," *ICC'93,* Geneva, Switzerland, May 23-26, 1993, pp. 910 - 914.

[14] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness,* W. H. Freeman, New York, 1979.

[15] D. W. Matula, G. Marble, and J. D. Isaacson, "Graph Coloring Algorithms," in *Graph Theory and Computing,* R. C. Read, editor, Academic Press, New York, 1972, 109-122.

[16] M. Duque-Antón, D. Kunz, and B. Rüber, "Channel Assignment for Cellular Radio Using Simulated Annealing," *IEEE Transactions on Vehicular Technology,* vol. 42, no. 1, Feb. 1993.

[17] M. Duque-Antón and D. Kunz," Channel Assignment Based on Neural Network Algorithms," in *Proc. DMR IV,* Oslo, Norway, June 26-28, 1990, pp. 5.4.1-5.4.9.

[18] M. Serizawa and D. J. Goodman, "Instability and Deadlock of Distributed Dynamic Channel Allocation," *1993 43rd IEEE Vehicular Technology Conference,* Secaucus, NJ, May 18-20, 1993.

[19] J. F. Whitehead, "Signal-Level-Based Dynamic Power Control for Co-channel Interference Management," *VTC'93,* Secaucus, NJ, May 18-20, 1993.

Design Options and Performance in Cellular Systems

Yiannis Argyropoulos and **Srikanta P.R. Kumar**
Electrical Engineering and Computer Science
Northwestern University
Evanston, Illinois 60208

Hazem Tawfik
Bell Northern Research
Richardson, Texas 75081

Abstract

We study the performance of a cellular network with respect to channel allocation, traffic variability and call quality. More specifically, the gain obtained by dynamic channel allocation is assessed, and the performance of both dynamic and fixed channel allocation is studied under non-uniform traffic. The blocking probability and call quality for the system are also obtained under non-uniform traffic conditions. The networking requirements for the interconnection of Base Stations are also addressed. We conclude this study with comparisons between different configurations of the system, and their effect on the above parameters.

1 Introduction

It has been shown that dynamic channel allocation yields lower blocking probability and better spectrum efficiency in cellular systems. In this study, we compare fixed and dynamic channel allocation schemes with respect to call quality and network load in addition to the above performance measures. The requirements for the backbone network are also assessed, together with the optimal number of Base Stations that have to be connected to a controller, as shown in Fig. 1. The controller is responsible for call set-up (including channel search), handoffs and other functions of the system, as registration and paging, and performs those without communication with the switching center.

2 Model

The system under study consists of a 8×8 cells, or 64 Base Stations. A sector of the simulation area, is shown in Fig. 1. The channel allocation technique is Purely Dynamic, in which it is assumed that Base Stations are equipped with radios tunable over the set of available frequencies, or Fixed, where only a predefined set of channels can be accessed from each Base Station. There are 35 radios per Base Station, and a total of 245 channels available (7-cell re-use pattern).

Incoming traffic is distributed uniformly or according to a pattern. Once a call is established, portables move according to one of the following patterns:

- In straight lines, with 4 possible directions (N,S,E and W),
- In a random direction, chosen uniformly between 0-360°.

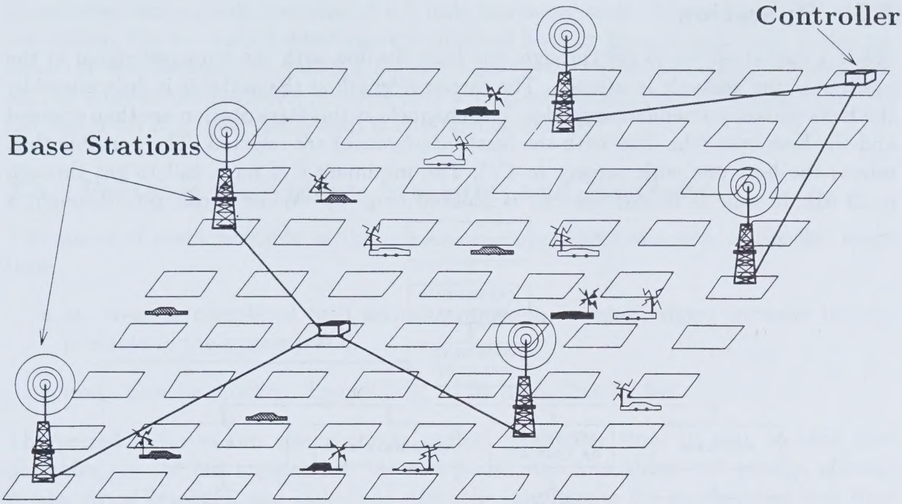


Figure 1: Simulation area

In the former case, there is a combination of classes of portables with different speeds, and the portables change direction when their average travelled distance is 300 m. In this study, we use 3 classes, moving at 0.5, 15 and 25 miles per hour, with percentages 20%, 40% and 40% respectively. In the first case, the distance covered before a change of direction is uniformly distributed between 0-60 feet,while for the other cases, the corresponding numbers are 0-600 feet. The status of an active portable is checked every time it covers a distance of 60 feet.

2.1 Traffic and Attenuation model

The population of portables in the service area is finite. The call origination per portable follows a Poisson process with average 2 calls per hour. The duration of a call is exponentially distributed with mean 100 seconds. Thus, each portable generates 0.055 Erlangs of traffic. For example, for a cell diameter of 0.5 mile, traffic volume of 20 Erlangs per Base Station approximately corresponds to 36.08 Erlangs/km^2.

The Hata attenuation model[1] has been used for the calculation of the signal strength of portables or Base Stations. Some sets of results have been obtained with Gaussian shadow fading added to the attenuation model. The results are obtained using omnidirectional antennas. In our model, only co-channel interference is considered, while no adjacent channels are used in the same cell. The background noise has been set to -128 dBm.

2.2 Operation

When a call attempts to get through, the Base Station with the strongest signal at the position of the portable is selected. The carrier strength at the portable is determined by the Hata distance-attenuation model. The channels at this Base Station are then scanned and the best ones (the ones with the least interference) are selected. The portable then selects the best one with respect to C/I. The minimum C/I for a call to get through is 20 dB. If none is found, the call is blocked (Fig. 2). When a call gets through, a transmitter power level is selected at the portable and Base Station so that the C/I of the call is above the target level of 20 dB. The possible power levels are 17,20,21.7 and 23 dBm, corresponding to 50, 100, 150, and 200 mW respectively. The maximum power is more than enough for the calls to maintain a C/I above the handoff level at the cell

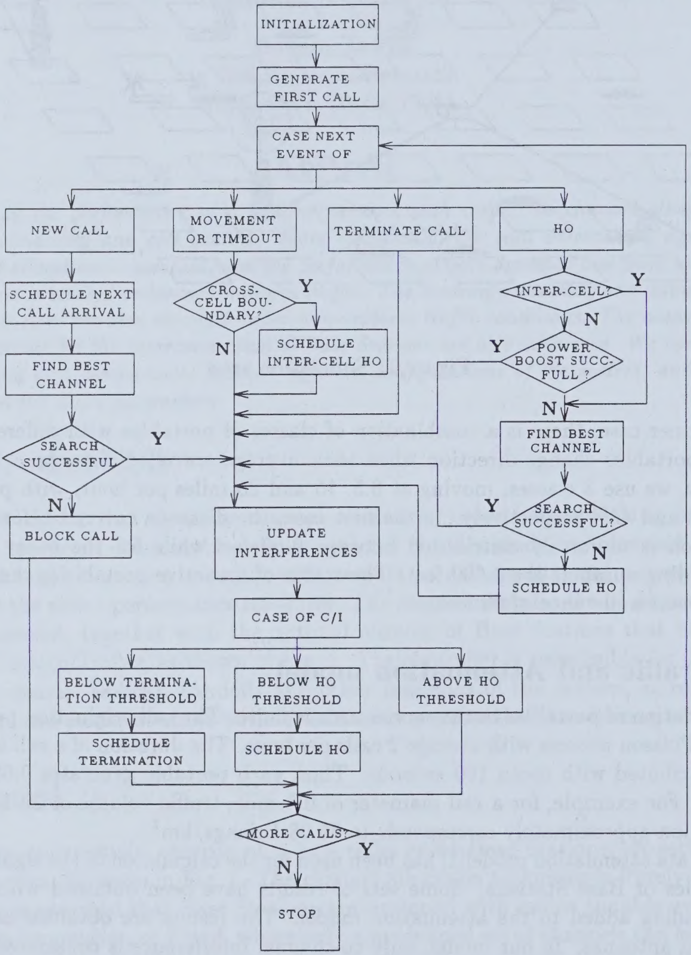

INITIALIZATION
GENERATE FIRST CALL
CASE NEXT EVENT OF
NEW CALL
MOVEMENT OR TIMEOUT
TERMINATE CALL
HO
SCHEDULE NEXT CALL ARRIVAL
CROSS-CELL BOUNDARY?
Y
N
SCHEDULE INTER-CELL HO
INTER-CELL?
Y
N
POWER BOOST SUCCFULL ?
Y
N
FIND BEST CHANNEL
SEARCH SUCCESSFUL
Y
N
FIND BEST CHANNEL
SEARCH SUCCESSFUL?
Y
N
BLOCK CALL
UPDATE INTERFERENCES
SCHEDULE HO
CASE OF C/I
BELOW TERMINATION THRESHOLD
BELOW HO THRESHOLD
ABOVE HO THRESHOLD
SCHEDULE TERMINATION
SCHEDULE HO
MORE CALLS?
Y
N
STOP

Figure 2: Flow Diagram of the Simulation Model

boundaries, since a cell diameter of 0.5 mile has been used. When a new call enters the system, the co-channel interference is updated both at Base Stations and portables. Since the specifications of AMPS[2],[3] have been used (where applicable), the channels are full-duplex, and portables interfere only with Base Stations and vice versa. Since the cell size is 0.5 of a mile, the system is interference limited.

2.3 Portable movement, Statistics and Handoffs

The status of every portable in the service area, regardless of speed, is checked every time

- an event is completed (call arrival-completion, handoff, signal increase for any portable in the system), and,
- every time its position changes (Fig. 2) by more than 60 feet.

The period between two status checks cannot be greater than 10 sec., so that slow portables are not left unattended. At this point, statistics about call quality, channel re-use, signal level etc. are collected. Also, the interference for all portables and Base Stations is updated and, if necessary, handoffs are scheduled.

By monitoring the portables, it is ensured that the total system interference remains up-to-date and the call quality remains above a pre-specified threshold. Furthermore, the statistics are taken after the system reaches a steady state and are within 5% of the mean with 95% confidence interval, computed using the Batch Means technique [4].

2.3.1 Handoffs

If the C/I of a call drops below a predefined threshold (17 dB throughout all the runs), a handoff process is initiated. If the portable has the capability of boosting the signal level, it attempts to improve the signal quality by doing so. In case this fails, and if the portable's closest Base Station is still the current one, the available channels at the current Base Station are searched for one that fulfills the C/I requirement. In case such a channel exists, an *intra-cell* handoff occurs. Otherwise, the handoff is blocked, and the portable may request another handoff at the end of the procedure, which may take up to 7 seconds (see[6] for a discussion). There is no time restriction between handoffs, since the system imposes a minimum time between handoff requests.

In case the portable detects a Base Station with a stronger signal, an inter-cell handoff occurs, following the same channel search procedure as in call initiation. It is thus guaranteed that portables are always locked on to the strongest Base Station. Another option is to let portables request a handoff when their C/I ratio drops below the handoff threshold regardless of whether they have crossed the cell boundary or not. It has been found from results not presented in this study, that the former handoff algorithm performs much better than the latter, especially at high loads.

If at any time during a call its C/I remains below 10 dB for 3 secs, then the call is dropped (Forced Termination). The completion time of the handoff is uniformly distributed between 3-7 sec. It has been found that this time is load dependent[5], but for simplicity, independence is assumed.

If the call quality falls below the termination threshold sooner than the handoff procedure is complete, the call is dropped. If, on the other hand, the call quality improves above the handoff threshold, the handoff is cancelled.

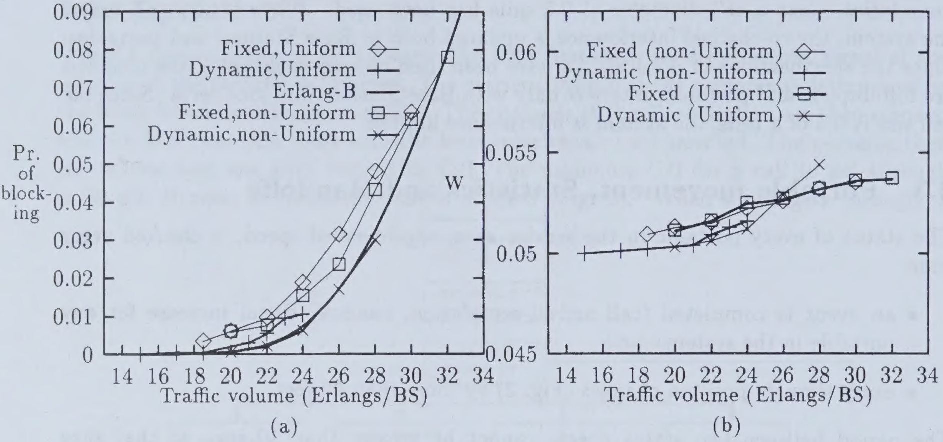


Figure 3: (a) Blocking probability. (b) Average transmitted power level.

3 Results

Two groups of results are presented in this study, for uniform and non-uniform traffic. Curves for both Dynamic and Fixed Channel allocation are obtained. The simulation area is roughly 4×4 miles. The non-uniform traffic pattern distributes 10% of the incoming traffic in a narrow strip of 400 feet width across the service area, simulating the effect of a main road or highway.

The results obtained correspond to blocking probabilities less than 10%, as shown in Fig. 3. As expected, fixed channel allocation performs worse than dynamic in the range of interest. In addition, the non-uniform traffic pattern, yields higher blocking probability compared to uniform. Both Dynamic and Fixed allocation curves are shifted upwards in the case of non-uniform traffic. In the latter case, dynamic channel allocation yields up to one-third of the blocking probability of fixed channel allocation, especially at low loads. The relative difference between fixed and dynamic channel allocation with respect to uniform and non-uniform traffic patterns, even though present, is not obvious. This is because only 10% of the total traffic is distributed according to a non-uniform pattern, and its effects are averaged out over the simulation area.

Dynamic channel allocation requires less transmitted power at the portable, as can be seen in Fig.3(b), which affects positively the portable's battery life. An interesting observation is that beyond 3% blocking probability, the dynamic channel allocation requires a higher power level. This is because the portables under dynamic channel allocation, beyond 26 Erl/Base Station perform a significantly higher number of handoffs (Fig. 4(b)) than with fixed channel allocation. Since the first step of the handoff procedure is to boost the power, the portables have on the average higher transmitted signal level. In general, the required power level at the portable increases at a very slow rate as the traffic increases, but for practical purposes, and for small blocking probabilities (less than 3%) may be considered independent of the traffic load.

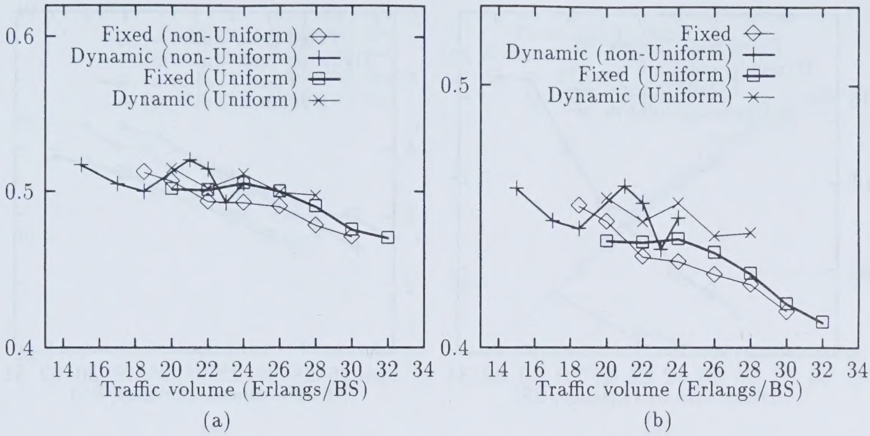


Figure 4: (a) Number of channel changes per call. (b) Average number of handoffs per call.

The number of handoffs per call for high loads and fixed channel allocation is significantly lower than that of dynamic channel allocation at the same loads (Fig. 4(a),(b)). This holds especially for channel changes within the same Base Station. This is not surprising, as fixed channel allocation has less interference variability than dynamic, and thus requires less channel arrangements.

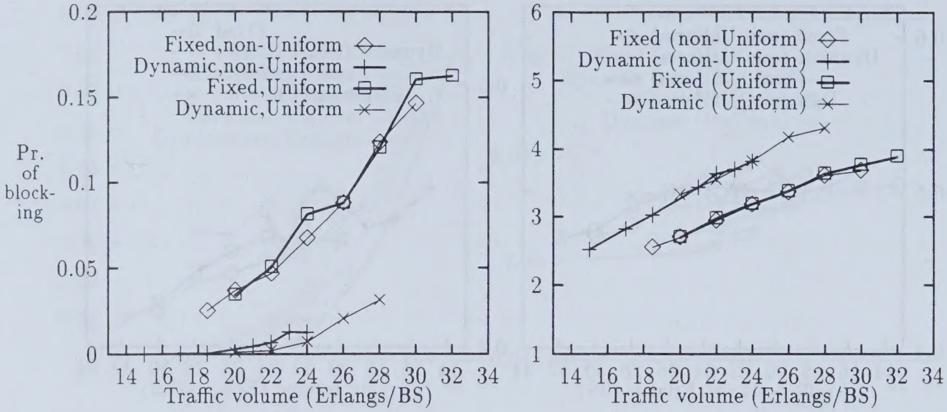
The number of handoffs per call drops slightly as traffic increases. This is partially due to the fact that the blocking probability of handoffs is higher at higher loads. This means that handoff requests have a high rejection rate, which is higher than the blocking probability, since a rejected handoff request will be repeated (Fig. 5(a)).

As it is expected, dynamic channel allocation has better channel packing than fixed (Fig. 5(b)), but the call quality is lower (6(a)). Equivalently, it is admitting more calls into the system (while the system constraints are not violated), at the expense of call quality.

In figures 7 and 8, we present the Base Station functions that the controller is required to perform.

4 Conclusions

We have studied the performance of a cellular system under uniform and non-uniform traffic for both dynamic and Fixed Channel allocation algorithms. Despite that dynamic channel allocation has slightly lower call quality than fixed, it seems the superior choice for a system. It leads to higher spectrum efficiency, less network load and higher capacity. Furthermore, its benefits are more evident in the case of non-uniform traffic, which better approximates a real system.



(a) (b)

Figure 5: (a) Blocking probability of handoffs. (b) Average number of times a channel is being re-used in the simulation area

References

[1] Masaharu Hata, "Empirical Formula for Propagation Loss in Land Mobile Radio Services", IEEE Transactions on Vehicular Technology, vol. VT-29, no 3, August 1980, pp. 317-325.

[2] FCC, "Code of Federal Regulations", part 22, Revised as of 10/1992, pp. 88-228.

[3] William C. Y. Lee, *Mobile Cellular Telecommunications Systems*, , McGraw-Hill.

[4] McMillan, *Engineering and Statistics*, , McGraw-Hill.

[5] Carlos Carpintero, Josep Paradells, Vicente Casares, and Javier Martinez, "Requirements for the Interconnection of Base Stations in a Wireless System", IEEE Proceedings, PCN Conference, 1992, pp. 141-146.

[6] A.O. Fapojuwo, A. McGirr and Saied Kazeminejad, "A Simulation Study of Speech Traffic Capacity in Digital Cordless Telecommunication Systems", IEEE Transactions on Vehicular Technology, vol. 41, no 1, February 1992, pp. 6-16.

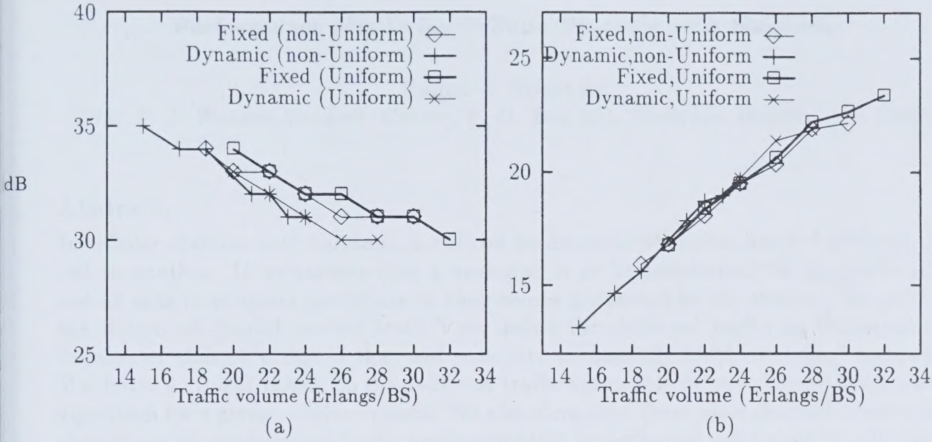


Figure 6: (a) Average call quality (C/I) in dB. (b) Average number of radios used at a Base Station

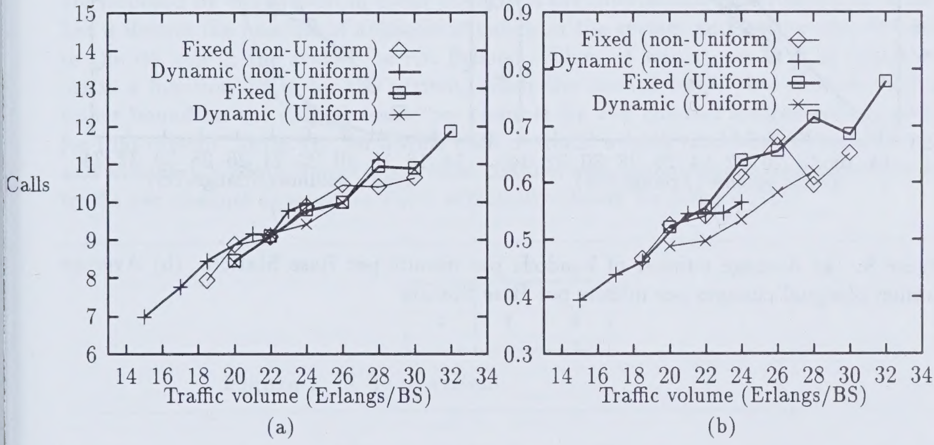


Figure 7: (a) Average number of calls per minute per Base Station. (b) Average number of channel changes per minute per Base Station

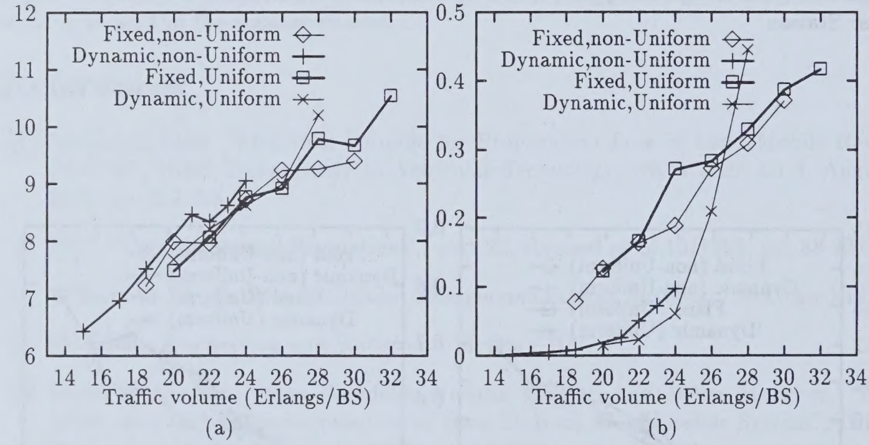


Figure 8: (a) Average number of handoffs per minute per Base Station. (b) Average number of signal changes per minute per Base Station

Performance Limits for Cellular Systems with Handoffs

Kumar N. Sivarajan
IBM T. J. Watson Research Center, P. O. Box 704, Yorktown Heights, NY 10598

Abstract.

In cellular systems with handoffs, a call can be dropped when it is handed off from one cell to another. If we assume that a customer is to be reimbursed for dropped calls, not all calls in progress contribute to the revenue generated by the system. To capture the notion of "useful carried traffic" we define the *delivered traffic* as the expected number of calls in progress that will complete successfully (without being dropped). We derive an upper bound on the delivered traffic achievable by any channel assignment algorithm for a given cellular system. We also show that there exist channel assignment algorithms whose delivered traffic approaches this upper bound arbitrarily closely when the number of available channels in the system is sufficiently large.

1. Introduction.

Consider a *static* cellular system, which we denote by S, where calls arrive at random times and have a random duration but originate and terminate in the same cell, i.e., there are no handoffs. A *channel assignment algorithm* must either assign a channel to a call that arrives or block it. In assigning a channel to a call, the channel assignment algorithm must obey certain restrictions on simultaneous uses of the same channel that are imposed by the system in order to control the interference experienced by a user. Let n denote the number of available channels in the system and let the offered traffic to the ith cell in the system be $r_i n$ Erlangs. Then, it was shown in [1,2] that there exists a function $T_S(r)$ for the system, called the *carried traffic function* which is an upper bound on the carried traffic per channel for *any* channel assignment algorithm for that system. Moreover, for a wide class of traffic models (including Poisson arrivals and arbitrary holding times) there exist channel assignment algorithms whose carried traffic per channel approaches $T_S(r)$ arbitrarily closely for large n [1,2].

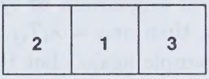


Figure 1. A 3-cell system.

Example. Consider the 3-cell system of Figure 1 and assume that there are *no handoffs*. In this system, the constraints on channel reuse are that the same channel may not be simultaneously used in two adjacent cells. Let the offered traffic be $\rho = nr/3$ Erlangs per cell. Using the results of [1,2] we can show the following. The carried traffic function for this system is $T_S(r) = \min\{r, r/3 + 1, 2\}$. A *fixed channel assignment algorithm* is one that allocates a fixed number of channels to each cell and is a greedy channel assignment algorithm in that it blocks a call in a cell if and only if all channels allocated to that cell are occupied. For $r \leq 3/2$, the carried traffic per channel of a fixed channel assignment algorithm that allocates $n/2$ channels to each of the cells approaches $T_S(r) = r$ arbitrarily closely for large n, i.e., this fixed channel assignment algorithm is asymptotically optimal for $r \leq 3/2$. Similarly, for $r \geq 3$, a fixed channel assignment algorithm that allocates all n available channels to each of the end-cells 2 and 3 is asymptotically optimal since its carried traffic approaches $T_S(R) = 2$. For every $r \in (3/2, 3)$ there exists a (different) fixed channel assignment algorithm that is similarly asymptotically optimal.

In this paper, we generalize the results of [1,2] to cellular systems with handoffs.

2. Traffic Model.

We assume there are T types of calls in the system each with a distinct *handoff sequence.* The handoff sequence of a call is the ordered list of cells that the call passes through, starting with the cell in which the call originates and ending with the cell in which the call terminates.

Example. Consider the 3-cell system and assume calls get handed off at most once (and to one of the neighbouring cells). Then, we have 7 types of calls in this system with handoff sequences: (1), (2), (3), (1, 2), (1, 3), (2, 1), and (3, 1).

If the handoff sequence of calls of type i, $i = 1, \ldots, T$, has length l_i, and $L = \max_i l_i$, the handoff sequences can be represented by a $T \times L$ matrix $H = (h_{ij})$ where h_{ij} is the jth element in the handoff sequence of calls of type i. ($h_{ij} = 0$, if $j > l_i$.) *

For the 3-cell example,

$$H = \begin{pmatrix} 1 & 0 \\ 2 & 0 \\ 3 & 0 \\ 1 & 2 \\ 1 & 3 \\ 2 & 1 \\ 3 & 1 \end{pmatrix}.$$

For static cellular systems, all handoff sequences have length 1.

The (normalized) *offered traffic* for the system is represented by a $T \times L$ matrix $R = (r_{ij})$ where nr_{ij}, $j \leq l_i$, is the expected number of type i calls that would be in progress in cell h_{ij} if *all* call requests were honored in *all* cells, and $r_{ij} = 0$, if $j > l_i$.

The offered traffic is related to the arrival rates of calls and their expected durations as follows. If λ_i denotes the rate of origination of type i calls in the system and T_{ij} their expected duration in cell h_{ij}, then $nr_{ij} = \lambda_i T_{ij}$.

Example. Consider the 3-cell example again. Let the rate of arrival of *new* calls be λ in each cell. Let 1/3 of the calls originating in cell 1 terminate there without being handed off, 1/3 get handed off to cell 2 and 1/3 to cell 3, i.e., $\lambda_i = \lambda/3$, $i = 1, 4, 5$. Let 1/2 the calls originating in each of cells 2 and 3 terminate there without being handed off and let the other 1/2 get handed off to cell 1, i.e., $\lambda_i = \lambda/2$, $i = 2, 3, 6, 7$. Further let the expected total duration of all calls be T and let the expected durations of a handoff call in the originating and terminating cells be equal to $T/2$, i.e., $T_{11} = T_{22} = T_{33} = T$

---

* If we assume that the number of handoffs experienced by a call is bounded by a constant D and the number of neighboring cells to which a call may be handed off is bounded by a constant Δ, then the number of handoff sequences is bounded by $N\Delta^D$. Therefore, the number of handoff sequences only grows linearly with the number of cells, N.

and $T_{ij} = T/2$, $i = 4,5,6,7$, $j = 1,2$. If $3\lambda T = nr$, the offered traffic (per channel) matrix for this example is

$$R = \begin{pmatrix} 1/9 & 0 \\ 1/6 & 0 \\ 1/6 & 0 \\ 1/18 & 1/18 \\ 1/18 & 1/18 \\ 1/12 & 1/12 \\ 1/12 & 1/12 \end{pmatrix} r.$$

Note that the total offered traffic is $n \sum_{ij} r_{ij} = nr$.

For any channel assignment algorithm A, the (normalized) *carried traffic* for the system is represented by a $T \times L$ matrix $C^{A,n} = (c_{ij}^{A,n})$ where $nc_{ij}^{A,n}$, $j \leq l_i$, is the expected number of type i calls that are in progress in cell h_{ij}, under the operation of channel assignment algorithm A, and $c_{ij}^{A,n} = 0$, if $j > l_i$. If we assume each call in progress generates revenue at a rate of unity (say, \$1 per minute), the (normalized) carried traffic corresponds to the expected rate of revenue per channel from the system.

In cellular systems with handoffs, the quantity of interest is not the carried traffic but the *delivered traffic*, which is defined as follows. For any channel assignment algorithm A, the (normalized) *delivered traffic* for the system is represented by a $T \times L$ matrix $D^{A,n} = (d_{ij}^{A,n})$ where $nd_{ij}^{A,n}$, $j \leq l_i$, is the expected number of type i calls that are in progress in cell h_{ij}, and that *will terminate without being dropped*, under the operation of channel assignment algorithm A, and $d_{ij}^{A,n} = 0$, if $j > l_i$. Essentially, in using the delivered traffic as the performance measure of a channel assignment algorithm, we are assuming that dropped calls do not earn any revenue. This is a reasonable assumption from the customer's point of view: he is reimbursed for all dropped calls.

For clarity, we will often write c_{ij} and d_{ij}, suppressing the dependence of the carried and delivered traffic on the channel assignment algorithm and the number of available channels, n.

The summary of the rest of this paper is as follows. We show that there is a function $T_S(R)$, which we call the delivered traffic function, for any cellular system which satisfies the model of Section 3 below which is an upper bound on the (normalized) delivered traffic of any channel assignment algorithm for that system. We also show that there exist channel assignment algorithms whose (normalized) delivered traffic approaches this upper bound arbitarily closely for sufficiently large n.

3. Cellular System Model.

Our model for the cellular system is almost the same as that of [1] and is as follows. We represent the state of an N-cell system by an N-vector $(m_1, \ldots, m_N)$ where m_k is the number of calls in progress in cell k. We denote the set of feasible states of the system when the number of available channels is n by S'_n. Note that if $s_n \in S'_n$ and $0 \leq s'_n \leq s_n$, then $s'_n \in S'_n$. We assume there is a convex subset S' of R^N which serves as an "unscaled model" for the system and which satisfies:

1. If $s \in S'$ and $0 \leq s' \leq s$, then $s' \in S'$.
2. If $s_n \in S'_n$, then $s_n/n \in S'$.
3. For every $s \in S'$, $\lfloor ns \rfloor \in S'_n$. †

---

† [1] assumes instead that for every $s \in S'$, there is a sequence (s_n) with $s_n \in S'_n$, such that $\lim_{n\to\infty} s_n/n = s$. All the results of this paper hold under this more general

For the 3-cell system, since the only restriction on channel reuse is that the same channel may not be used simultaneously in adjacent cells, it can be shown that,

$$S'_n = \{m = (m_1, m_2, m_3) : m \geq 0,\ m_1 + m_2 \leq n,\ m_1 + m_3 \leq n\},$$

and

$$S' = \{t = (t_1, t_2, t_3) : t \geq 0,\ t_1 + t_2 \leq 1,\ t_1 + t_3 \leq 1\}.$$

4. Indistinguishability of Call Types.

We define the prefixes of the handoff sequence h_i, $l_i > 1$, as the (handoff) sequences h_k, of length $l_k \leq l_i$ such that, $h_{kj} = h_{ij}$, $j = 1, \ldots, l_k$. If h_k is a prefix of h_i, we write $h_k \preceq h_i$ and say h_i is an *extension* of h_k. In the 3-cell example, $(1) \preceq (1,2)$. (Note that a handoff sequence is a prefix of itself.) We *assume* that all prefixes of a handoff sequence are also handoff sequences.

We assume that the system does not have any knowledge of the "future" of a call; it only knows the handoff "history" of a call. Thus, it cannot distinguish between calls of type i and k in cells $h_{k,j} = h_{i,j}$, $j = 1, \ldots, l_k$, if h_i is an extension of h_k. In the 3-cell example, the system cannot distinguish among type (1), type $(1,2)$ and type $(1,3)$ calls in cell 1.

Under the operation of a channel assignment algorithm A, let $nc_{ij}^{A,n} = \lambda_{ij}^{A,n} T_{ij}$, i.e., $\lambda_{ij}^{A,n}$ is the arrival rate of accepted calls of type i in cell h_{ij}. Then the indistinguishability property, which we assume holds axiomatically, can be stated as follows.

Indistinguishability Property. *Under the operation of any channel assignment algorithm A,*

$$\frac{\lambda_{kj}^{A,n}}{\lambda_k} = \frac{\lambda_{ij}^{A,n}}{\lambda_i}, \qquad j = 1, \ldots, l_k, \qquad \text{if } h_k \preceq h_i.$$

This property states that since calls of type i and k cannot be distinguished in cell h_{ij} (h_{kj}), the fraction of accepted (blocked) calls of these two types must be equal for any channel assignment algorithm.

We also observe that the following property holds.

Conservation of Call Arrivals. *Under the operation of any channel assignment algorithm A,*

$$\lambda_i \geq \lambda_{i1}^{A,n} \geq \lambda_{i2}^{A,n} \geq \ldots \geq \lambda_{i,l_i}^{A,n}.$$

The inequalities are due to blocked/dropped calls.

The following lemmas follow from these two properties and the definitions of carried and delivered traffic.

Lemma 1 states that the carrying probabilties, c_{*j}/r_{*j}, of types of calls that are indistinguishable in a cell must be equal in that cell.

Lemma 1. *For any channel assignment algorithm,*

$$\frac{c_{kj}}{r_{kj}} = \frac{c_{ij}}{r_{ij}}, \qquad j = 1, \ldots, l_k, \qquad \text{if } h_k \preceq h_i.$$

---

model as well.

Proof. This lemma follows from the indistinguishability property using $nr_{ij} = \lambda_i T_{ij}$ and $nc_{ij} = \lambda_{ij} T_{ij}$. (This can also be viewed as an alternative statement of the indistinguishability property in terms of traffic instead of arrival rates.) ■

The following lemma relates the delivered traffic to the carried traffic. It states that the delivery probability d_{ij}/r_{ij}, of a type of call (in a cell), equals its carrying probability, c_{il_i}/r_{il_i} in the last cell of its handoff sequence.

Lemma 2. *For any channel assignment algorithm,*

$$d_{ij} = \frac{r_{ij}}{r_{i,l_i}} c_{i,l_i}, \qquad j = 1, \ldots, l_i.$$

Proof.

$$\begin{aligned} d_{ij}/r_{ij} &= \lambda_{i,l_i} T_{ij} / \lambda_i T_{ij} \qquad \text{(definition of } d_{ij}\text{)} \\ &= \lambda_{i,l_i} T_{i,l_i} / \lambda_i T_{i,l_i} \\ &= c_{i,l_i} / r_{i,l_i}. \end{aligned}$$

■

Lemma 3. *For any channel assignment algorithm, the carried traffic satisfies the following prefix condition:*

$$\frac{c_{k,l_k}}{r_{k,l_k}} \geq \frac{c_{i,l_i}}{r_{i,l_i}}, \qquad \text{if } h_k \preceq h_i.$$

Proof.

$$\begin{aligned} \frac{c_{k,l_k}}{r_{k,l_k}} &= \lambda_{k,l_k} T_{k,l_k} / \lambda_k T_{k,l_k} \\ &= \lambda_{i,l_k} / \lambda_i \qquad \text{(indistinguishability of } h_k \text{ and } h_i \text{ in cell } l_k\text{)} \\ &\geq \lambda_{i,l_i} / \lambda_i \qquad \text{(call conservation)} \\ &= \lambda_{i,l_i} T_{i,l_i} / \lambda_i T_{i,l_i} \\ &= c_{i,l_i} / r_{i,l_i}. \end{aligned}$$

■

Lemma 3 states that the delivery probability of a call-type cannot be more than the delivery probability of any of its prefixes. In the 3-cell example, this states that the delivery probability of type (1, 2) calls cannot be more than those of type (1) calls.

5. Performance Limits: Upper Bound.

We will need a few definitions, which we will illustrate using the 3-cell example, before we are ready to state the main results of our paper.

Let

$$U(R) = \left\{ s = (s_1, s_2, \ldots, s_T) : 0 \leq s_i \leq r_{i,l_i}, \right. \\ \left. \frac{s_k}{r_{k,l_k}} \geq \frac{s_i}{r_{i,l_i}}, \quad \text{if } h_k \preceq h_i \right\}.$$

For the 3-cell example,

$$U(R) = \left\{ s = (s_1, \ldots, s_7) : 0 \leq s \leq (1/9, 1/6, 1/6, 1/18, 1/18, 1/12, 1/12) r, \right. \\ \left. s_1 \geq 2s_4,\ s_1 \geq 2s_5,\ s_2 \geq 2s_6,\ s_3 \geq 2s_7 \right\}.$$

The significance of the space $U(R)$ is stated by the following lemma.

Lemma 4. *For any channel assignment algorithm,*

$$c = (c_{1,l_1}, c_{2,l_2}, \dots, c_{T,l_T}) \in U(R).$$

Proof. Since the carried traffic is non-negative and at most the offered traffic, $0 \leq c_{i,l_i} \leq r_{i,l_i}$. Combined with Lemma 3, we have,

$$c \in U(R).$$ ∎

Let

$$q_k = \sum_{i:h_k \preceq h_i} r_{i,l_k}.$$

nq_k is the offered traffic of type k and its extensions in cell h_{k,l_k}. For the 3-cell system,

$$q = (2/9, 1/4, 1/4, 1/18, 1/18, 1/12, 1/12)r.$$

Let

$$p_k^{A,n} = \sum_{i:h_k \preceq h_i} c_{i,l_k}^{A,n}.$$

$np_k^{A,n}$ is the carried traffic of type k and its extensions in cell h_{k,l_k}.

The p_k's are related to the q_k's as follows.

Lemma 5.

$$p_k = q_k c_{k,l_k}/r_{k,l_k}.$$

Proof.

$$\begin{aligned} p_k r_{k,l_k} &= \sum_{i:h_k \preceq h_i} c_{i,l_k} r_{k,l_k} \qquad \text{(definition of } p_k\text{)} \\ &= \sum_{i:h_k \preceq h_i} r_{i,l_k} c_{k,l_k} \qquad \text{(using Lemma 1)} \\ &= q_k c_{k,l_k} \qquad \text{(definition of } q_k\text{).} \end{aligned}$$ ∎

Let

$$S = \left\{ s : t_k = \sum_{i:h_{i,l_i}=k} q_i s_i / r_{i,l_i},\ t \in S' \right\}.$$

For the 3-cell system,

$$\begin{aligned} S &= \{s:\ t_1 = 2s_1 + s_6 + s_7,\ t_2 = 3s_2/2 + s_4,\ t_3 = 3s_3/2 + s_5,\ t \in S'\} \\ &= \{s \geq 0:\ 2s_1 + 3s_2/2 + s_4 + s_6 + s_7 \leq 1,\ 2s_1 + 3s_3/2 + s_5 + s_6 + s_7 \leq 1\}. \end{aligned}$$

The significance of the space S is stated by the following lemma.

Lemma 6. *For any channel assignment algorithm,*

$$c = (c_{1,l_1}, c_{2,l_2}, \ldots, c_{T,l_T}) \in S.$$

Proof. Under the operation of any channel assignment algorithm, let m_i denote the (random) number of calls in progress in cell i. Consider an arbitrary cell k that is part of handoff sequence j but is not the last cell in this handoff sequence, i.e., $h_{j\alpha} = k$, but $\alpha < l_j$. Since we assume that every prefix of a handoff sequence is also a handoff sequence, there is some other handoff sequence i of length $\alpha = l_i$ such that $h_i \preceq h_j$. Thus, $h_{i,l_i} = k$, i.e., cell k is the last cell in handoff sequence i. Therefore, we can compute the carried traffic in cell k by considering each traffic type, i, for which cell k is the last cell in its handoff sequence and summing over the carried traffic of that type and its extensions (in cell k), as

$$\begin{aligned} E(m_k) &= n \sum_{i:h_{i,l_i}=k} p_i \\ &= n \sum_{i:h_{i,l_i}=k} q_i c_{i,l_i}/r_{i,l_i} \qquad \text{(by Lemma 5).} \end{aligned}$$

Also, since $m \in S'_n \subseteq nS'$ and S' is convex, $E(m) \in nS'$. Therefore, by definition of S, $c \in S$. ∎

Let

$$T_S(R) = \max\left\{\sum_{i=1}^{T} r_i s_i/r_{i,l_i} \ : \ s \in S \cap U(R)\right\},$$

where $r_i = \sum_j r_{ij}$, be the *delivered traffic function* for the cellular system S. For the 3-cell example,

$$\begin{aligned} T_S(R) &= \max\{s_1 + s_2 + s_3 + 2s_4 + 2s_5 + 2s_6 + 2s_7 : s \in S \cap U(R)\} \\ &= \min(r, r/6 + 6/5, 8/5). \end{aligned}$$

Let $T_S^{A,n}(R) = \sum_{ij} d_{ij}$ be the delivered traffic for the channel assignment algorithm A operating on the system S with the offered traffic matrix R and n available channels. Then the delivered traffic function $T_S(R)$ has the following significance.

Theorem 1. *For any channel assignment algorithm A,*

$$\frac{1}{n} T_S^{A,n}(R) \le T_S(R) \qquad \textit{for } R \ge 0.$$

Proof. For the channel assignment algorithm A, let $c = (c_{1,l_1}, c_{2,l_2}, \ldots, c_{T,l_T})$. By lemmas 4 and 6, $c \in S \cap U(R)$.

$$\begin{aligned} T_S^{A,n}(R) &= n \sum_{ij} d_{ij} \\ &= n \sum_i \frac{c_{i,l_i}}{r_{i,l_i}} \sum_j r_{ij} \qquad \text{(by Lemma 2)} \\ &\le n \max\left\{\sum_i r_i s_i/r_{i,l_i} : s \in S \cap U(R)\right\} \qquad \text{(since } c \in S \cap U(R)) \\ &= n T_S(R). \end{aligned}$$

∎

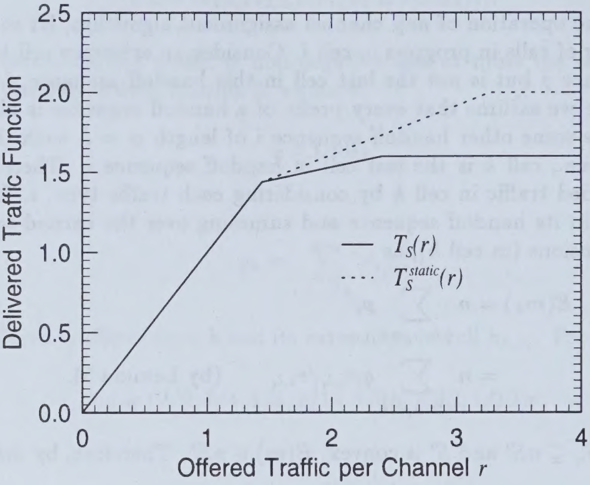


Figure 2. The delivered traffic functions $T_S(r)$ and $T_S^{\mathrm{static}}(r)$ for the 3-cell system of Figure 1.

In the "static" case, Theorem 1 reduces to Theorem 3.1 of [1].

Consider the 3-cell example. Note that the offered traffic (of all types) in cell 1 is $7nr/18$ and in cells 2 and 3 is $11nr/36$. If the offered traffic were the same and there were no handoffs, ($H^T = (1,2,3)$ and $R^T = (7/18, 11/36, 11/36)r$) the delivered/carried traffic function for this system would be

$$T_S^{\mathrm{static}}(r) = \min(r, 11r/36 + 1, 2).$$

For the 3-cell example, the functions $T_S(r)$ and $T_S^{\mathrm{static}}(r)$ are plotted in Figure 2. Note that $T_S(r) = T_S^{\mathrm{static}}(r) = r$ for $r \leq 36/25$ and both $T_S(r) < r$ and $T_S^{\mathrm{static}}(r) < r$ for $r > 36/25$, in agreement with Theorem 7.1 of [1].

6. Asymptotic Tightness of Upper Bound.

So far we have left the statistics of the call arrivals and holding times largely unspecified. We shall show that the upper bound of Theorem 1 is asymptotically tight for traffic models for which new calls are independent of one another and that satisfy the following Asymptotic Traffic Property (ATP) [2] just as in the static case [1,2].

Consider a sequence of single cell systems (indexed by n) with offered traffic r_n Erlangs, number of available channels x_n and a greedy channel assignment algorithm that blocks a call only if no channel is available to carry that call. Let $\tau_n(r_n, x_n)$ denote the carried traffic for the nth one-cell system under the greedy channel assignment algorithm. The traffic statistics are said to satisfy the ATP if

$$\lim_{n\to\infty} \frac{\tau_n(r_n, x_n)}{n} = \min(r, x)$$

for arbitrary sequences r_n and x_n where $\lim_{n\to\infty} r_n/n = r$ and $\lim_{n\to\infty} x_n/n = x$.

The ATP simply states that if the offered traffic is more than the number of available channels, then asymptotically all the available channels are occupied and if the offered traffic is less than the number of available channels, then asymptotically the fraction of blocked/dropped calls is zero.

If $l_k > 1$, let $n\varrho_k^{A,n}$ denote the offered traffic of type k and its extensions in cell h_{k,l_k} under the operation of the channel assignment algorithm A, i.e.,

$$n\varrho_k^{A,n} = \sum_{i:h_k \preceq h_i} \lambda_{i,l_k-1}^{A,n} T_{i,l_k}.$$

(Recall that $\lambda_{ij}^{A,n}$ is the arrival rate of accepted calls of type i in cell h_{ij} under the operation of channel assignment algorithm A.)

Lemma 7. *If $h_i \preceq h_k$ and $l_k = l_i + 1$, for any channel assignment algorithm,*

$$\varrho_k = p_i q_k / q_i.$$

Proof. From Lemma 5, $p_i/q_i = c_{i,l_i}/r_{i,l_i}$. Therefore, it suffices to show that $n\varrho_k r_{i,l_i} = nq_k c_{i,l_i}$.

$$\begin{aligned}
n\varrho_k r_{i,l_i} &= \left(\sum_{j:h_k \preceq h_j} \lambda_{j,l_k-1} T_{j,l_k}\right)(\lambda_i T_{i,l_i}) \\
&= \sum_{j:h_k \preceq h_j} \lambda_{j,l_i} T_{j,l_k} \lambda_i T_{i,l_i} \qquad \text{(using } l_k - 1 = l_i\text{)} \\
&= \sum_{j:h_k \preceq h_j} \lambda_{i,l_i} T_{i,l_i} \lambda_j T_{j,l_k} \\
&\qquad \text{(by indistinguishability of } h_j \text{ and } h_i \text{ in cell } l_i,\ \lambda_{j,l_i}\lambda_i = \lambda_{i,l_i}\lambda_j) \\
&= nc_{i,l_i} q_k.
\end{aligned}$$

∎

Lemma 8. *Let $\hat{s} \in S \cap U(R)$. Then, there exists a fixed channel assignment algorithm F that allocates $\lfloor nq_i\hat{s}_i/r_{i,l_i}\rfloor$ channels to cell h_{i,l_i} for calls of type i and its extensions. Moreover, if the traffic statistics satisfy the ATP,*

$$\lim_{n\to\infty} p_i^{F,n} = q_i\hat{s}_i/r_{i,l_i}.$$

Proof. Under the fixed channel assignment algorithm F, the number of channels allocated to cell k is,

$$f_k = \sum_{i:h_{i,l_i}=k} \lfloor nq_i\hat{s}_i/r_{i,l_i}\rfloor \le \left\lfloor \sum_{i:h_{i,l_i}=k} nq_i\hat{s}_i/r_{i,l_i} \right\rfloor = f_k'.$$

By the definition of S, since $\hat{s} \in S$, the vector $\hat{t} \in S'$ where

$$\hat{t}_k = \sum_{i:h_{i,l_i}=k} q_i\hat{s}_i/r_{i,l_i}.$$

Therefore, by assumption 3 of Section 3, $f'_k \in S'_n$ and hence, $f_k \in S_n$. Thus, the fixed channel assignment algorithm F is feasible.

To compute $\lim_{n\to\infty} p_i^{F,n}$, we use induction on l_i, the length of call-type i. For $l_i = 1$,

$$\begin{aligned}\lim_{n\to\infty} p_i^{F,n} &= \lim_{n\to\infty} \frac{1}{n}\tau(nq_i, \lfloor nq_i\hat{s}_i/r_{i,l_i}\rfloor) \\ &= \min(q_i, q_i\hat{s}_i/r_{i,l_i}) \qquad \text{(using the ATP)} \\ &= q_i\hat{s}_i/r_{i,l_i} \qquad \text{(since } \hat{s} \in U(R),\ \hat{s}_i \le r_{i,l_i}\text{)}.\end{aligned}$$

For $l_i > 1$, let $h_k \preceq h_i$ and let $l_k = l_i - 1$.

$$\begin{aligned}\lim_{n\to\infty} p_i &= \lim_{n\to\infty} \frac{1}{n}\tau(n\varrho_i^{F,n}, \lfloor nq_i\hat{s}_i/r_{i,l_i}\rfloor) \\ &= \lim_{n\to\infty} \frac{1}{n}\tau(np_k^{F,n}q_i/q_k, nq_i\hat{s}_i/r_{i,l_i}) \qquad \text{(by Lemma 7)} \\ &= \min((q_k\hat{s}_k/r_{k,l_k})q_i/q_k, q_i\hat{s}_i/r_{i,l_i}) \qquad \text{(induction hypothesis and the ATP)} \\ &= q_i\hat{s}_i/r_{i,l_i} \qquad \text{(since } \hat{s} \in U(R),\ \hat{s}_k/r_{k,l_k} \ge \hat{s}_i/r_{i,l_i}\text{)}.\end{aligned}$$

∎

The following theorem generalizes Theorem 4.1 of [1] to cellular systems with handoffs.

Theorem 2. *If the traffic statistics satisfy the ATP, there exists a channel assignment algorithm F, such that,*

$$\lim_{n\to\infty} \frac{1}{n} T_S^{F,n}(R) = T_S(R),$$

where $T_S^{F,n}(R)$ is the delivered traffic for the channel assignment algorithm F.

Proof. Let $\hat{s} \in S \cap U(R)$. Consider a fixed channel assignment algorithm F that allocates $nq_i\hat{s}_i/r_{i,l_i}$ channels to cell h_{i,l_i} for calls of type i and its extensions. For this channel assignment algorithm,

$$\begin{aligned}T_S^{F,n}(R) &= n\sum_{ij} d_{ij}^{F,n} \\ &= n\sum_i c_{i,l_i}^{F,n} r_i/r_{i,l_i} \qquad \text{(using Lemma 2 and } \textstyle\sum_j r_{ij} = r_i\text{)} \\ &= n\sum_i (p_i^{F,n} r_{i,l_i}/q_i) r_i/r_{i,l_i} \qquad \text{(using Lemma 5)}.\end{aligned}$$

Therefore,

$$\lim_{n\to\infty} \frac{1}{n} T_S^{F,n}(R) = \sum_i \hat{s}_i r_i/r_{i,l_i} \qquad \text{(using Lemma 8)}.$$

Since this is valid for arbitrary $\hat{s} \in S \cap U(R)$ we can choose $\hat{s} \in S \cap U(R)$ to maximize $\sum_i \hat{s}_i r_i/r_{i,l_i}$. For this particular choice of $\hat{s}$,

$$\lim_{n\to\infty} \frac{1}{n} T_S^{F,n}(R) = T_S(R).$$

∎

Consider the 3-cell example and a fixed, greedy channel assignment algorithm that allocates $3n/5$ channels to each of the end-cells (2 and 3) and $2n/5$ channels to the central cell (1) with the restriction that the channels in the end-cells are to be used only for new calls arising in that cell and of the $2n/5$ channels allocated to the central cell, $n/5$ are to be used exclusively for calls handed off from cell 2 and $n/5$ for calls handed off from cell 3. All calls originating in cell 1 are blocked. If $r \geq 12/5$ and if the traffic statistics satisfy the ATP, the fraction of dropped calls will be zero asymptotically and both the carried and delivered traffic for this system approach $8n/5$ which is the best achievable delivered traffic by any channel assignment algorithm for this system (from Theorem 1).

The reader should compare this channel assignment algorithm with the fixed channel assignment algorithm for the static system that is asymptotically optimal for $r \geq 36/11$. This channel assignment algorithm allocates all n available channels to each of the end-cells and no channels to the central cell. The carried traffic for this channel assignment algorithm does approach $2n$ *even when there are handoffs* but the *delivered traffic* only approaches $4n/3 < 8n/5$.

7. Comments and Conclusions.

Note that the fixed channel assignment algorithm of Theorem 2 that is asymptotically optimal for large n is impractical, even if n is large, because it is a function of R. We do not know if there exist channel assignment algorithms, in general, which can be implemented without knowing the offered traffic R and which are asymptotically optimal for every value of R.

As a final point, instead of the delivered traffic, we could maximize any linear function of the carried traffic. Thus we could assign different rates of revenues to different types of calls as was done in [1]. We could also maximize the carried traffic itself which would correspond to assuming that the customer is not refunded for dropped calls (he pays for dropped calls for the time used). For this case, the *carried traffic function* for the 3-cell example is

$$T_S^{\text{carried}}(R) = \min(r, 11r/36 + 1, r/5 + 6/5, 2)$$

which is bounded below by the delivered traffic function $T_S(R)$ and above by the delivered/carried traffic function for the corresponding static system $T_S^{\text{static}}(R)$. It is interesting to note that this function is different from both of them.

Acknowledgements.

I thank Rajiv Ramaswami for his comments on improving the presentation of this paper.

References.

[1] McEliece, R. J., and K. N. Sivarajan, "Maximizing Marginal Revenue in Generalized Blocking Service Networks," *Proc. 1992 Allerton Conference on Communication, Control and Computing.*

[2] McEliece, R. J., and K. N. Sivarajan, "Performance Limits for Channelized Cellular Telephone Systems," to appear in *IEEE Trans. Inform. Theory.* (Earlier versions appeared in *Proc. 1990 Allerton Conference on Communication, Control and Computing*, pp. 869–880 and *Proc. Third IEE Conference on Telecomm.*, Edinburgh, pp. 360–365.)

Control of Partially Observed Discrete Event Systems with Variable Lookahead Maximal Policies

Nejib Ben Hadj-Alouane [*] , Stéphane Lafortune [†] and Feng Lin [‡]

Abstract

This paper deals with the on-line control of partially observed discrete event systems (DES's). The goal is to restrict the system's behavior within a prefix-closed specification while accounting for the presence of *uncontrollable* and *unobservable* events. The off-line synthesis of control policies for partially observed DES is of exponential complexity (in the worst case) in the number of states of the system, say $|X|$. In the spirit of recent work on the on-line control of partially observed DES [6] and on variable lookahead control of fully observed DES [5], we propose an on-line approach where following each observable event, a control action is computed using an algorithm of linear complexity in $|X|$ (again in the worst case). In contrast to the method of Heymann and Lin, our algorithm has the following properties: (1) no off-line calculations (such as finding the supremal controllable sublanguage of the legal language) are necessary; (2) the algorithm always generates a *maximal* observable and controllable sublanguage of the legal language; (3) hence, if the *supremal* observable and controllable sublanguage exists, the algorithm will automatically generate it; and (4) by proper parameter selection, any maximal observable and controllable sublanguage of the legal language can be generated. The algorithm can also be modified so as to generate a maximal observable and controllable sublanguage of the legal language that is guaranteed to contain the supremal normal and controllable sublanguage.

Keywords: supervisory control, partial observation, variable lookahead policies, on-line control, forward search, maximal observable and controllable sublanguage

1 Background on Supervisory Control

First, we overview the standard supervisory control problem – SSCP – described in detail in [9]. Then, we outline the corresponding partial observation version introduced in [8].

The discrete event system is modeled by a generator (or state machine) $G = (\Sigma, X, \delta, x_0, X_m)$, where Σ denotes the events, and X denotes the state space. The transition function, $\delta : X \times \Sigma \longrightarrow X$, describes system dynamics: Given states $x, y \in X$ and an event $\sigma \in \Sigma$, $\delta(x, \sigma) = y$, if the execution of σ from state x takes the system to state y. The system has a unique initial state, x_0, and X_m is a special subset of X, the subset of marked states (to model completed operations, for example).

We use the extended version of δ, also denoted by δ (for simplicity), and defined from $X \times \Sigma^*$ into X: Given states $x, y \in X$ and a *trace* $t \in \Sigma^*$, $\delta(x, t) = y$ if starting at x, it is possible to execute the sequence of events specified by t, and this execution moves the system to y.

The state machine G characterizes two languages, $L(G)$ and $L_m(G)$. The set of traces *generated* by G is the language

$$L(G) \overset{def}{=} \{s \in \Sigma^* \ : \ \delta(x_0, s) \text{ is defined } \}.$$

---

[*] Address: Department of Electrical Engineering and Computer Science; University of Michigan; Ann Arbor, MI 48109-2122; (313)936-2830 – nejib@eecs.umich.edu.

[†] Research supported in part by the National Science Foundation under grant ECS-9057967 with additional support from DEC and GE. Address: Department of Electrical Engineering and Computer Science; University of Michigan; Ann Arbor, MI 48109-2122; (313)763-0591 – stephane@eecs.umich.edu.

[‡] Research supported in part by the National Science Foundation under grant ECS-9213922. Address: Department of Electrical and Computer Engineering; Wayne State University; Detroit, MI 48202; (313)577-3428 – flin@ece.eng.wayne.edu.

The language $L(G)$ is prefix closed since it equals its prefix closure, denoted $\overline{L(G)}$, and defined as the set of all prefixes of the traces in $L(G)$.

The set of traces *marked* by G is

$$L_m(G) \overset{def}{=} \{s \in \Sigma^* \ : \ \delta(x_0, s) \in X_m\}.$$

In this paper, we only deal with prefix-closed languages; i.e. we assume that $L_m(G) = L(G)$ (or $X = X_m$).

In the SSCP framework, a prefix-closed subset of $L(G)$, K, is referred to as the *specification* (the "legal", allowable or desired part of the behavior of G). A *supervisor* (or controller) is imposed on G with the task of dynamically enabling/disabling events in order to "achieve" K. The supervisor's task is further complicated by the partitioning of Σ into two subsets: Σ_c, the subset of *controllable* events (those that can be enabled/disabled by the supervisor) and Σ_{uc}, the subset of *uncontrollable* events.

Formally, a supervisor or control policy is a function

$$\gamma \ : \ L(G) \longrightarrow 2^{\Sigma},$$

with the following constraint: $(\forall s \in L(G))(\Sigma_u \subseteq \gamma(s))$. The prefix-closed behavior that results from imposing γ on G is denoted $L(G, \gamma)$ and recursively defined by

(1) $\epsilon \in L(G)$,

(2) $(\forall s \in L(G, \gamma), \sigma \in \Sigma)(s\sigma \in L(G, \gamma) \Leftrightarrow \sigma \in \gamma(s))$.

Ideally, γ is such that $L(G, \gamma) = K$. In practice, however, such γ may not exist; but there is an "optimal" alternative. This is formally characterized by the following definition.

Definition 1 *A prefix-closed sublanguage K is* ***controllable*** *if,*

$$(foralls \in K, \sigma \in \Sigma_{uc})(s\sigma \in L(G) \Rightarrow s\sigma \in K)$$

It is easy to see that only controllable subsets (sublanguages) of K can be achieved by means of the control method outlined above. In [10] it is shown that the *supremal controllable sublanguage* of K, the optimal solution, denoted by $K^{\uparrow}$, always exists. Moreover, there are algorithms for computing γ's that achieve $K^{\uparrow}$, when $L(G)$ and K are given by *finite* state machines. An on-line algorithm which can provide the control action for a single state at a time (the state just entered by the system), by examining only a portion of $G \times H$ (where H is the finite state machine generating K) is provided in [5] and reviewed in the next section.

Under the partial observation assumption of [8], Σ is also partitioned into the subset of observable events, Σ_o, and the subset of unobservable events, Σ_{uo}. The projection, $P \ : \ \Sigma^* \longrightarrow \Sigma_o^*$ defined by:

$$P(\epsilon) = \epsilon \text{ and } (\forall s \in \Sigma^*, \sigma \in \Sigma) P(s\sigma) = \begin{cases} P(s)\sigma & \text{if } \sigma \in \Sigma_o \\ P(s) & \text{if } \sigma \in \Sigma_{uo}, \end{cases}$$

characterizes the behavior observed by the supervisor, namely $P(L(G))$.

A partial observation supervisor is a function

$$\gamma_p \ : \ P(L(G)) \longrightarrow 2^{\Sigma},$$

with the following constraint: $(\forall s \in P(L(G)))(\Sigma_{uc} \subseteq \gamma_p(s))$. The behavior, $L(G, \gamma_p)$, generated by the above supervisor is defined by,

(1) $\epsilon \in L(G, \gamma_p)$,

(2) $(\forall s \in L(G, \gamma_p), \sigma \in \Sigma)(s\sigma \in L(G, \gamma_p) \Leftrightarrow \sigma \in \gamma_p(P(s)))$.

The property of observability defined below is important in characterizing the existence of partial observation supervisors.

Definition 2 *A prefix closed sublanguage K is* ***observable*** *if,*

$$(\forall s, s' \in K, \sigma \in \Sigma)(P(s) = P(s') \wedge s\sigma \in K \wedge s\sigma \in L(G)) \Rightarrow s'\sigma \in K.$$

For a given specification K, a partial observation supervisor generating K exists if and only if K is observable and controllable [8]. Unfortunately, if K does not satisfy the above condition, there is not an optimal sublanguage that could be generated instead; i.e. the supremal controllable and observable sublanguage need not exist. In general, there are many maximal controllable and observable sublanguages; this paper is concerned with computing them on-line.

The notion of normality (a strong version of observability), defined below, is important in partial observation supervision since there is a supremal controllable and normal sublanguage.

Definition 3 *A prefix closed sublanguage K is* ***normal*** *if,*

$$(\forall s \in L(G))(P(s) \in P(K) \Leftrightarrow s \in K)$$

The supremal controllable and normal sublanguage is often offered as a solution to partial observation supervision problems (see, e.g. [8, 4]). In this paper, we improve on this situation by providing an algorithm that computes maximals containing the supremal controllable and normal sublanguage.

2 Variable Lookahead Control with Full Observation

Variable lookahead control of fully observed systems and with state information was most recently discussed in [5]. Under this approach, control actions are determined as needed on-line and by exploring only a portion of the state space of the system using a forward search algorithm. Below, we review from [5] the aspects that are utilized in this paper, namely the **VLP-S** algorithm. (The more general problem of limited lookahead on-line control, and with no state information, is discussed in [1, 3, 2].)

From now on, we assume (without loss of generality) that the specification language K is generated by a submachine of G; if this assumption is not satisfied a-priori, the submachine of G can be refined appropriately. Thus, we can talk of the legality or illegality of states of X rather than traces of $L(G)$. The set of legal states of X is denoted by X_H.

The following cost function V (which can be partial) is used to record the relevant information obtained from computing all previously issued control actions:

$$V : X \longrightarrow \{0, \infty\}.$$

In the case of prefix closed languages, dealt with here, an infinite cost is assigned to a state if and only if it uncontrollably leads (i.e., through a trace of uncontrollable events) to an illegal state. The optimal policy γ for SSCP is defined in terms of V as follows.

- If $V(\delta(x_0, s)) = 0$ then $\gamma(s) = \{\sigma \in \Sigma_G(\delta(x_0, s)) \ : \ V(\delta(x_0, s\sigma)) = 0\}$.

- If $V(\delta(x_0, s)) = \infty$ then $\gamma(s) = (\Sigma_u \cap \Sigma_G(\delta(x_0, s)))$.

In the above, $\Sigma_G(x)$ denotes the active set of G at state x.

Each time the system executes a new event, a new control action needs to be determined and issued. For this purpose, values of V which are needed and are yet undefined are computed. The following algorithm **VLP-S** (for *variable lookahead policies* with *state* information) is used to efficiently perform this task for prefix-closed languages (the general algorithm is presented in [5]).

```
PROCEDURE   VLP-S($x_0 \in X$) :

1.   $Exp := \emptyset$;  Expand($x_0$);
     $D := Dom(V)$;
2.   $(\forall y \in ((Exp - X_H) - D))V(y) := \infty$;
3.   WHILE  $[(\exists y \in Exp - D)(\exists \sigma \in \Sigma_u)(V(\delta(y, \sigma)) = \infty)]$  DO $V(y) := \infty$;
4.   FOR $[y \in Exp - Dom(V)]$ DO $V(y) := 0$;
5.   RETURN;
```

Step 1 of **VLP-S** calls the procedure **Expand**, which is not shown here for brevity purposes. **Expand** returns in the variable **Exp** the state space of a submachine of G rooted at x. This submachine is expanded in a tree-like manner and includes the states accessible from x that may need to be examined in order to determine the cost of x (for closed languages, this is at most the uncontrollable reach of x). In steps 2 and 3, the submachine is trimmed by removing the states that are illegal and states that cannot be prevented (by control) from reaching illegal states.

The worst case complexity of an execution of **VLP-S** is linear in the number of expanded states It is important to note that (1) all expanded states are assigned costs, and (2) states of known costs are not expanded. Hence, the combined effort required to determine the cost of every state in X also has a linear complexity.

The following theorem, which we state without proof for brevity purposes, provides additional insight as to the usage of **VLP-S** and addresses its correctness (it combines theorem 1 and lemma 2 of [5]).

Theorem 1 $\sigma \in \Sigma \wedge s \in K^{\uparrow} \wedge V(\delta(x_0, s\sigma)) = 0 \Leftrightarrow s\sigma \in K^{\uparrow}$

3 Variable Lookahead Control with Partial Observation

Under the assumption of full observation, the controller can monitor every event that is executed by the system; hence a new control action can be computed and issued following each event occurrence. Here we consider SSCP with partial observation assumption introduced in [8] and [4]. Under this assumption, a controller can monitor only a subset of the events (the observable events). Also, the control action can only be updated following an observed event; i.e., in between consecutive observable events (while only unobservable events are being executed), the system operates under a constant control action.

In the on-line control paradigm, control actions are determined as needed: Almost instantaneously following the execution of an observable event the next control action is computed and

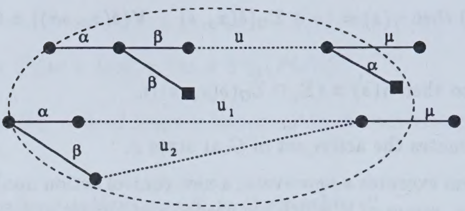


Figure 1: Unobservable Reach

issued. Since we only deal with closed languages (i.e., blocking is not an issue), intuitively, determining the next control action simply involves ensuring that the execution of the next observable event as well as any unobservable events in between cannot possibly allow the system to generate illegal behavior (either directly or via an uncontrollable trace). Offhand, this could require examining every possible unobservable continuation (as allowed by the uncontrolled system) of the currently observed behavior, the so-called *current unobservable reach* (formally defined below). Still, since the supremal controllable and observable sublanguage of a given language does not always exist [8], there is no unique and universal way of determining the next control action.

The approach presented in [6] uses a simple strategy for determining the next control action: Any event whose execution following a trace in the unobservable reach results in illegal behavior (of course, either immediately or through an uncontrollable extension) is disabled.

The above strategy is often too restrictive. The reason is that disabling an event prevents the occurrence of any continuation of the current behavior involving the given event, at least until the next control action is applied; hence, it is unnecessary to disable any additional events for the sole purpose of preventing the above mentioned continuations from resulting in illegal behavior. The above situation is illustrated in Figure 1. The states in the current unobservable reach are enclosed in the ellipse (though not all are shown); the small squares represent illegal states. Disabling the unobservable β, for example, automatically prevents the continuation, $u\alpha$, from occurring; hence, there is no need to further disable α (as would be required by the strategy of [6]). Surprisingly, it is shown in [6] that the behavior resulting from the above strategy always includes the supremal normal and controllable sublanguage of the specification language, the typical solution offered by existing off-line schemes.

Mutually exclusive choices may be involved in the selection of the control action; and this is the reason why the supremal controllable and observable sublanguage of the specification language does not always exist. In the situation of Figure 1, the prevention of illegal behavior (at least until the next control action) could be achieved by disabling α instead of β. Moreover, the resulting behaviors (i.e., the one corresponding to the disablement of β and the one corresponding to the disablement of α) are incomparable with respect to set inclusion (for example, the first excludes the continuation β which is included by the second).

Our on-line approach is based upon two main components. The first is an efficient method for keeping track of the current unobservable reach (it resembles the one used in [6]). The second is a strategy, inspired by the above two observations, for computing the next control action. The main tool for implementing our approach is the so-called *variable lookahead* algorithm for computing control *policies* under *partial observation* –**VLP-PO**– presented below.

```
PROCEDURE    VLP-PO($\beta_o \in \Sigma_o \cup \{\epsilon\}$);

1.  N-S := $\{x \in X \,:\, x = \delta(y, \beta_o)$, for $y \in$ P-S $\}$;
```

```
2.  Control-Action(N-S, Event-Ordering);
3.  P-S := $\{x \in X : x = \delta(y,u)$, for $y \in$ N-S and $u \in (\text{Act} \cap \Sigma_{uo})^*\}$;
4.  END;
```

The first component of our approach works as follows. Following the execution of an observable event **VLP-PO** is called to compute the next control action (to be issued immediately), which is returned in the global variable **Act**. The only other global variable **P-S** is used to store the set of *possible states* the system could be in prior to the most recent **VLP-PO** call. Step 1 of the call **VLP-PO(β_o)** computes **N-S**, the set of possible *next states* the system could enter immediately following the execution of the last observable event β_0; a state is a member of **N-S** if it can be reached via a β_o transition from a state in the most recent **P-S**. Step 2 calls the procedure **Control-Action** given below, which computes the next control action and stores it in **Act**. Step 3 updates the value of **P-S**, based on the current value **N-S** and the next control action stored in **Act**; the new value of **P-S** is the set of all states reachable from those in **N-S** via unobservable traces consisting only of events in **Act**.

The **initial conditions** for **VLP-PO** are:

- **P-S** := $\{q_0\}$ prior to the first call to **VLP-PO**, where q_0 is the initial sate of G,
- **Event-Ordering** := a list reflecting the desired ordering for the events of Σ_c, and in the desired order,
- **VLP-PO(ϵ)** is used to determine the first control action.

The value of **N-S** which passed in Step 2 of **VLP-PO** to the procedure **Control-Action**, is used to determine the so-called *extended unobservable reach*, which we define before listing **Control-Action**.

The so-called *unobservable reach* of a given set of states $S \subseteq X$ with respect to a given subset of events $E \subseteq \Sigma$, denoted $UR_E(S)$, is defined as follows,

$$UR_E(S) \overset{def}{=} \{x \in X \ : \ x = \delta(y,u) \text{ for } y \in S \text{ and } u \in (\Sigma_{uo} \cap E)^*\}.$$

The extended unobservable reach of S with respect to a subset of events E, denoted $UR{+}_E(S)$, consists of all the states in $UR_E(S)$ together with any states reachable from them via a single observable transition with an event in E. Formally,

$$UR{+}_E(S) \overset{def}{=} UR(S) \cup \{x \in X \ : \ x = \delta(y,\beta) \text{ , for } y \in UR(S) \text{ and } \beta \in (\Sigma_o \cap E)\}.$$

Following Step 2 of a given execution of **VLP-PO**, UR_Σ(**N-S**) is referred to as the current unobservable reach (since **N-S** is the current set of next states).

PROCEDURE Control-Action($S \subseteq X$, E-List : Ordered-Σ_c);

```
1.    Act := $\Sigma_{uc}$;  L-Pt := 1;
2.    WHILE [ E-List $\neq \emptyset$ ] DO
2.1   IF [ L-Pt > | E-List |) ] THEN
          Act := Act $\cup$ E-List;  RETURN;
2.2   IF [ $(\forall x \in UR{+}_{\text{Act}}(S))(\delta(x,$E-List.L-Pt$)$ is not defined) ] THEN
            L-Pt := L-Pt +1;  GOTO 2;
2.3   FOR [ $x \in UR{+}_{\text{Act}\cup\{\text{E-List.L-Pt}\}}(S)$ ] DO
          VLP-S($x$);
```

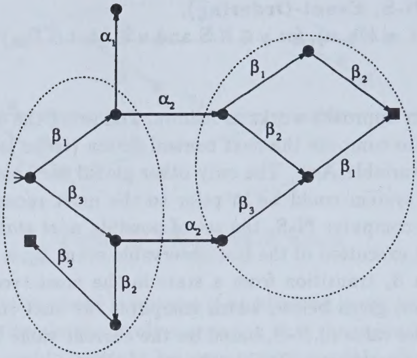


Figure 2: Machine

```
        IF  [ $V(x) = \infty$ ] THEN
            E-List := E-List - { E-List.L-Pt };  GOTO 2;
    2.4     Act := Act $\cup$ { E-List.L-Pt };  E-List := E-List $-$ { E-List.L-Pt };  L-Pt := 1;  GOTO 2;
    3.  END;
```

The strategy for computing the next control action uses two main rules. The first involves building the next control action gradually, by examining the controllable events one by one in accordance to a pre-specified ordering (any ordering is allowed). For each event, a check is made to ensure that enabling it, in addition to the events already enabled, would not incur illegal behavior from the current extended unobservable reach (obviously, only traces involving the new event together with the events already enabled need to be checked). The event is excluded from the next control action if this check fails. With the use of this rule, the situation of mutually exclusive events, discussed above, is totally avoided. The second rule involves the other precondition for including an event in the next control action: The event must add to the already enabled behavior in the current unobservable reach. This rule ensures the maximality of the resulting behavior.

The procedure **Control-Action** implements the above strategy as follows. With the event ordering given in the argument **E-List** Step 2.2 executes the second rule, and then Step 2.3 executes the first. In Step 2.4 the event currently under examination is permanently included in the control action, after satisfying the two preconditions given by the rules. Note that if the precondition of the second rule is not satisfied the event is not discarded and may be examined another time, when the enabled behavior in the current unobservable reach changes by including another event in the control action.

The following example illustrates the use of **VLP-PO** and provides a concrete motivation for the above strategy.

Example 1 *Consider the finite state machine in figure 2; its initial state is distinguished with an arrow and. The illegal states of the system are represented with small squares. Assume that all events are controllable. Moreover, the events labeled with α are observable while the ones labeled with β are unobservable.*

*The variable **P-S** is initialized to the singleton containing the initial state and the call **VLP-PO(ϵ)** is issued to compute the first control action. The current set of next states (the value of **N-S** following Step 1 of **VLP-PO**) includes only the initial state. The left most ellipse marks* *the current unobservable reach.*

Consider using the following ordering for the controllable events: The β's are placed first in accordance with their subscripts, the α's follow, also in accordance with their subscripts. The event β_1 is the first to be enabled. When examining β_2 for the precondition of the second rule fails (enabling it does not add a new trace to the current enabled extended unobservable reach consisting of the only trace β_1) hence it is skipped. The event β_3 is the second to be enabled. When β_2 is re-examined it fails the test of the first rule (it extends the trace β_3 to $\beta_3\beta_2\beta_3$, which leads to an illegal state) and is rejected. The events α_1 and α_2 are the last to be enabled. Hence, the first control action is $\{\beta_1, \beta_3, \alpha_1, \alpha_2\}$. The set of possible states as a result of this control action consists of the two states reachable by the traces β_1 and β_3. Note that the above control action cannot be improved and in fact it is the best possible control action.

*Let the next observable event executed by the system be α_2. Then the call **VLP-PO(α_2)** computes the second control action. The current next set of states contains the two states reachable by $\beta_1\alpha_2$ and $\beta_3\alpha_2$. By maintaining the previous ordering, we get the following control action $\{\beta_1, \alpha_1, \alpha_2\}$. Interchanging the positions of β_1 and β_2 within the same ordering results in a different control action, $\{\beta_2, \beta_3, \alpha_1, \alpha_2\}$. Note that both control actions cannot be improved, as **VLP-PO** always generates a maximal behavior.*

Since by definition, uncontrollable events must be part of any control action (Step 1 of **VLP-PO** initializes **Act** to Σ_u), a control policy γ_p generating a non-empty controllable subset of K may not always exist: Sometimes illegal behavior cannot be avoided at the initial state even by disabling all controllable events. In such situations, a starting error (or STE) is said to occur [1] (there is no first control action that guarantees the validity of $L(G,\gamma)$; hence, the system cannot be safely *started*). In the case of on-line control with full observation, the absence of an STE is contingent upon a nonempty supremal controllable sublanguage [1] (recall that the empty set is always controllable by convention). The following proposition provides the analogous condition for the case of on-line control with partial observation.

Proposition 2 *There is no STE if and only if the empty set is not a maximal controllable and observable sublanguage of K.*

The following result states that **VLP-PO** is correct.

Theorem 3 *If there is no STE then for each control policy γ computed using **VLP-PO**, $L(G,\gamma)$ is a controllable and observable sublanguage of K.*

Obviously, in the above theorem, the condition that an STE does not occur does not present a restriction on the applicability of **VLP-PO**; as we pointed out earlier, when an STE occurs, there is no control policy that can generate a controllable and observable sublanguage.

The last proposition of this section deals with the worst case computational complexity of **VLP-PO**.

Proposition 4 *In the worst case, the effort involved in computing each control action using **VLP-PO** is of the order $O(|X|)$.*

4 Maximality of VLP-PO Policies

In this section we state the main properties of our approach. All proofs are omitted for brevity purposes and will be presented elsewhere.

The following two results address the performance of the control policies produced using **VLP-PO**.

Theorem 5 *Let γ be a policy computed using* ***VLP-PO****. If there is no STE, then $L(G,\gamma)$ is a* ***maximal*** *observable and controllable sublanguage of K.*

Corollary 6 *If there is no STE and if the supremal observable and controllable sublanguage of K exists, then it is generated by every policy computed using* ***VLP-PO****.*

The next result basically states that every maximal observable and controllable sublanguage can be generated using **VLP-PO** (by appropriately setting the controllable event's ordering presented to **Control-Action** at every step).

Theorem 7 *Let K_m be a non-empty maximal controllable and observable sublanguage of K. Then for each $s \in P(L(G))$ there exists an ordering $\leq_s$, such that the policy computed using* ***VLP-PO*** *with these orderings generates K_m.*

Inline with the considerable attention given in the literature to the computation of the supremal controllable and normal sublanguage, we present a modified version of **VLP-PO** that generates maximal controllable and observable sublanguages which include the supremal controllable and normal sublanguage.

The new algorithm is called **VLP-PON**, with the "N", obviously, referring to "normal". Its only difference from **VLP-PO** is that it selects a controllable event's ordering which guarantees that the generated maximal contains the supremal controllable and normal sublanguage.

```
PROCEDURE    VLP-PON($\beta_o \in \Sigma_o \cup \{\epsilon\}$);

1.  N-S := $\{x \in X : x = \delta(y,\beta_o)$, for $y \in$ P-S $\}$;
2.  Control-N-Action(N-S);
3.  Control-Action(N-S, [ACT;$\Sigma_c-$ ACT]);
4.  P-S := $\{x \in X : x = \delta(y,u)$, for $y \in$ N-S and $u \in (\text{ACT} \cap \Sigma_{uo})^*\}$;
5.  END;
```

Steps 2 and 3 perform the selection of the ordering. The procedure **Control-N-Action**, presented below, is called in Step 2 to return in **Act** a control action which is larger than the one assigned under the policy γ_π of [6]. In step 3 the desired ordering is formed by placing the events enabled by the control action returned in Step 2 first. This ordering guarantees that the final control action returned in Step 3, by the call to the procedure **Control-Action**, generates a behavior that is larger than the behavior generated by the initial control action (actually the final control action does not have to enable every event enabled by the original one; some enabled events do not contribute in generating the behavior and may be disabled). The supremal controllable and normal sublanguage is included in $L(G,\gamma_\pi)$ (see [6]; also, a formula characterizing $L(G,\gamma_p i)$ is given in [7]); hence we show that it is also included in the sublanguage generated by the **VLP-PON** policy.

The supremal controllable and normal sublanguage can sometimes be included in more than one maximal (if there are several). All such maximal sublanguages could be generated by appropriately setting the order of the events disabled by the initial control action (Σ_c- ACT, in Step 3).

The procedure **Control-N-Action** is given below. It uses the same technique as the algorithm of [6], described in the previous section, to produce a control action larger than the one given under the policy generating the supremal controllable and normal sublanguage: In Step 3 an event is enabled only if whenever it is allowed from a state in the unobservable reach, it leads to to a zero cost state.

PROCEDURE Control-N-Action($S \subseteq X$);

```
1. ACT := $\Sigma_{uc}$;
2. FOR [ $x \in UR +_{\Sigma} (S)$ ] DO VLP-S($x$);
3. FOR [ $\sigma \in \Sigma_c$ ] DO
       IF [ $(\forall x \in UR_{\Sigma}(S)) V(\delta(x,\sigma)) = 0$ ] THEN
     ACT := ACT $\cup \{\sigma\}$;
4. END;
```

The following result states more formally the purpose of **VLP-PON**.

Theorem 8 *Let γ be a policy computed using **VLP-PON**. If there is no STE, then $L(G,\gamma)$ is a maximal controllable and observable sublanguage that is larger than the supremal controllable and normal sublanguage of K.*

References

[1] S. L. Chung, S. Lafortune, and F. Lin. Limited lookahead policies in supervisory control of discrete event systems. *IEEE Trans. Automatic Control*, 37(12):1921–1935, December 1992.

[2] S. L. Chung, S. Lafortune, and F. Lin. Recursive computation of limited lookahead supervisory controls for discrete event systems. *Journal of Discrete Event Dynamic Systems: Theory and Applications*, 3(1):71–100, March 1993.

[3] S. L. Chung, S. Lafortune, and F. Lin. Supervisory control using variable lookahead policies. In *Proc. 1993 American Control Conf.*, pages 1203–1208, San Francisco, CA, June 1993.

[4] R. Cieslak, C. Desclaux, A. Fawaz, and P. Varaiya. Supervisory control of discrete-event processes with partial observations. *IEEE Trans. Automatic Control*, 33(3):249–260, March 1988.

[5] N. B. Hadj-Alouane, S. Lafortune, and F. Lin. Variable lookahead supervisory control with state information. Technical Report CGR-93-9, College of Engineering Control Group Reports, University of Michigan, July 1993.

[6] M. Heymann and F. Lin. On-line control of partially observed discrete event systems. CIS Report 9310, Technion - Israel Institute of Technology, Haifa, March 1993.

[7] R. Kumar. Observability formulas for discrete event dynamical systems. In *Proc. 1993 Conf. Information Sciences and Systems*, Johns Hopkins University, March 1993.

[8] F. Lin and W.M. Wonham. On observability of discrete–event systems. *Information Sciences*, 44(3):173–198, 1988.

[9] P.J. Ramadge and W.M. Wonham. The control of discrete event systems. *Proc. of the IEEE*, 77(1):81–98, January 1989.

[10] W. M. Wonham and P. J. Ramadge. On the supremal controllable sublanguage of a given language. *SIAM J. Control and Optimization*, 25(3):637–659, May 1987.

Dynamics of a Switched Server System under Corridor Policies

Guo-Xiang Yu and Pirooz Vakili *
Department of Manufacturing Engineering
Boston University

Abstract

It has been shown that discretely controlled continuous variable systems can exhibit a variety of dynamic behaviors. In [1] and [8], it is shown that the trajectory of a continuous flow system can be eventually periodic or chaotic. In this paper, we study a three part-type manufacturing flow system under corridor policies. Both open and closed systems are considered. We give conditions on the parameters of the corridor that guarantee periodic or chaotic behavior.

1 Introduction

The use of fluid models to approximate discrete part manufacturing systems has proved to be very fruitful (see, e.g., [5], [7], [9], [2]) . Within this framework planning and scheduling problems can be addressed using more powerful analytical tools and the dynamic behavior of the system can be analysed with more facility. Solutions obtained from a fluid model approximation can be used to "guide" the operation of a discrete-part system, or to provide qualitative insight into the dynamic behavior of such systems. A number of authors have studied the dynamics of fluid models – or the discrepency between the fluid models and the discrete-part systems guided by them – and have pointed to possibilities of instability–in spite of sufficient capacity– and of complex dynamics such as chaotic behavior (see, e.g., [7], [8], [2], [1], [4]). The work in this paper is motivated by this direction of research, in particular the work of Sharifnia, Caramanis and Gershwin [8]. From a more abstract point of view, the fluid approximations can be viewed as non-linear dynamical systems operating under discrete –often event-driven– control feedback policies (see, e.g., [1]). We benefit from this point of view and, in fact, while we are motivated by problems considered in [8], the examples we consider in this paper are cast in the framework of switched server systems considered in by Chase, Serrano and Ramadge in [1].

---

*The work in this paper was supported in part by the National Science Foundation under Grants EID-9212122 and DDM-9215368.

In [8], Sharifnia et al. considered a corridor policy for the control of the set-up switching times of a fluid system used as a model of multi-part-type manufacturing system with set-ups. They showed that under some conditions on the parameters of the corridor the sequence of set-ups visited is periodic, and experimentally observed that for some parameters of the corridor the system exhibits chaotic behavior. Motivated by this problem, we consider a switched server system operating under corridor policies. In this case the server switches buffers when the trajectory meets a face of a linear corridor. We give conditions on the parameters of the corridor that guarantee periodic or chaotic behavior.

This rest of the paper is organized as follows. In section 2, we describe the general model of the system and the corridor policy. In section 3, we derive results for closed switched server systems. The results for open systems are given in Section 4. We conclude in Section 5.

2 Model description

Consider a single server processing work (fluid) that arrive at constant rates $\rho_1, \cdots, \rho_N$ to N buffers. The following assumption are in force: processing rate of the server is assumed to be 1, the server can process work from only one of the buffers at any instant in time, and the location of the server is a control variable. We also assume that there is no time delay to switch between buffers, i.e., the setup time is zero.

Let $x_i(t)$ be the workload in buffer i at time t and let the buffer state be $\mathbf{x}(t) = (x_1(t), \cdots, x_N(t))$. The server state is defined by $I(t) = (I_1(t), \cdots, I_N(t))$ where $I_i(t) = 1$ if machine is processing buffer i at time t; $I_i(t) = 0$ otherwise. The dynamic equation for the buffer state is

$$\dot{\mathbf{x}}(t) = \rho - I(t), \tag{1}$$

where $\rho = (\rho_1, \cdots, \rho_N)$. Let $\mathbf{v}_i = \rho - \mathbf{e}_i$ be the velocity of change in $\mathbf{x}$ associated with when the server is serving buffer i, ($\mathbf{e}_i$ is the unit vector in the ith-axis direction). Assume the server processes buffer i during the time interval $[t_0, t_1]$ during which time the buffer is non-empty; then

$$\mathbf{x}(t) = (t - t_0)\mathbf{v}_i + \mathbf{x}(t_0) \quad t \in [t_0, t_1]$$

in other words, the buffer state dynamics is linear on this interval.

Consider a linear corridor C in $\mathbf{R}^N$ where the vertex of the corridor is placed at the origin. The corridor is characterized by N vectors in $\mathbf{R}^N$, $\{\mathbf{n}_1, \ldots, \mathbf{n}_N\}$; $\mathbf{n}_i$ is the normal vector to the ith face of the corridor. The corridor determines the feedback control policy for switching between buffers as follows: once the trajectory reaches the ith face of the corridor, the server switches to the ith buffer and the buffer state moves in the direction of $\mathbf{v}_i$ until the next face of the corridor is reached and so on. The dynamics of this system is completely determined by the velocities $\{\mathbf{v}_i; 1 \le i \le N\}$ and the normals to the faces of the corridor, $\{\mathbf{n}_i; 1 \le i \le N\}$. We assume that the velocities are fixed and are interested in understanding system dynamics when the parameters of the corridor, i.e., $\mathbf{n}_i$'s change.

We assume that $\sum_{i=1}^{N} \rho_i \le 1$. Consider the case $\sum_{i=1}^{N} \rho_i < 1$. In this case the processing capacity of the server is strictly higher than the offered total workload and it can be easily shown that the buffer trajectory converges to zero; in this sense the system dynamics is well understood and is in fact trivial. However, in the course of convergence to zero, infinite number of switches among buffers occur. It is this sequence of setup changes (more explicitly, the sequence of buffers visited) that is of interest to us.

More precisely, let $F : C \to C$ be the one step map defined on the corridor, i.e., $F(\mathbf{x})$ is the next point on the corridor if the processing begins when the buffer state is $\mathbf{x}$. Consider an initial state $\mathbf{x}_0$. Let $\mathbf{x}_n = F^n(\mathbf{x}_0)$ and $s_n = i$ if $\mathbf{x}_n.\mathbf{n}_i = 0$, i.e., if the point $\mathbf{x}_n$ is on the ith face of the corridor. The switching sequence $\{s_n, n \ge 0\}$ is one of the subjects of our study. It is possible to simplify the problem by realizing that for all initial points lying on a line going through the origin the switching sequence is the same. To formalize, we define the following equivalence relation on $C - \{0\}$:

$$\mathbf{x} \sim \mathbf{y} \quad \text{iff} \quad \mathbf{x} = t\mathbf{y} \quad \text{for some } t \in \mathbf{R}$$

This equivalence relation simply says that all points that lie on a line passing through the origin are equivalent. Let $T = C/\sim$, i.e., T is the quotient space of C with respect to the equivalence $\sim$. Let

$$\pi : C \to T$$

be the projection from C to T; i.e. $\pi(\mathbf{x}) = [\mathbf{x}]$ where $[\mathbf{x}]$ is the equivalence class of $\mathbf{x}$. The one step mapping F projects down to a mapping $G : T \to T$, i.e., it is easy to verify that

Lemma: Given the above assumptions:

$$F \circ \pi = \pi \circ F$$

$(F([x])$ is defined as $[F(x)])$.

Consider any hyperplane in $\mathbf{R}^N$ that does not pass through the origin. The intersection of this hyperplane with the corridor can be taken as a representation of T and the sequence of buffers visited, $\{s_n, n \ge 0\}$, can be studied on this intersection.

In the case when $\sum_{i=1}^{N} \rho_i = 1$, i.e., a closed switched server system, the buffer trajectory satisfies $\sum_{i=1}^{N} \mathbf{x}_i(t) = \sum_{i=1}^{N} \mathbf{x}_i(0)$ and hence it stays within a set similar to that described above.

The case of $N = 3$ is the smallest dimension for which the above problem is non-trivial and can be analysed with relative ease.In addition, in this case a range of qualitatively different dynamic behaviors is observable. We limit ourselves to $N = 3$. We begin with a closed switched server system which, as we will make clear, is the simpler of the two cases of $\sum_{i=1}^{N} \rho_i = 1$, and $\sum_{i=1}^{N} \rho_i = 1$.

Remark. Our use of the term switched server system is somewhat different from that used by Chase et al. [1]. The switched arrival system in their terminology becomes one example of a switched server system (our terminology) under a corridor policy. We find the terminology of switched server system more useful in our context. Sharifnia et al. [8] assume the server is satisfying a constant demand for each buffer and assume that there is a backlog of demand. Under the projection defined above both cases, the backlogged demand and the positive inventory of work, can be analysed in an identical framework.

3 Closed Switched Server System

Let the triangle ABC be the intersection of the corridor with the plane defined by $\sum_{i=1}^{3} \mathbf{x}_i(t) = \sum_{i=1}^{3} \mathbf{x}_i(0)$. Select a two dimensional coordinate system such that the vertex A is at the origin and the side AB is placed on the x-axis. Modifying the notation of the previous section, let $\mathbf{n}_i$, $i = 1, 2, 3$, be the unit length normals of the three sides of triangle ABC which point toward the inside of the triangle. Let l_i be the side associated with buffer i, i.e., when the trajectory reaches l_i the server switches to the ith buffer. Let $\mathbf{v}_i$ be the processing velocity vector if the server is processing buffer i (see Figure 1). We assume that the following conditions hold:

$$\mathbf{n}_i . \mathbf{v}_i > 0; \quad i = 1, 2, 3 \tag{2}$$

With this assumption the trajectory $\mathbf{x}(t)$ will be confined to the triangle ABC. Let G be the one step map defined on the triangle ABC, let $\{\mathbf{x}_n; n \geq 0\}$ be the sampled sequence of buffer states (defined by $\mathbf{x}_n = G^n(\mathbf{x}_0)$), and let $\{s_n; n \geq 0\}$ be the sequence of buffers visited (defined by $s_n = i$ if $\mathbf{x}_n \in l_i$).

Theorem 1 *If*

$$\mathbf{v}_1 . \mathbf{n}_3 \geq 0, \quad \mathbf{v}_2 . \mathbf{n}_1 \geq 0, \quad \mathbf{v}_3 . \mathbf{n}_2 \geq 0, \tag{3}$$

then $\{s_n,\ n \geq 1\}$ *is a round-robin switching sequence.*

Proof: Assume the trajectory starts from side 1. Since $\mathbf{v}_1 . \mathbf{n}_3 \geq 0$, it can not hit side 3 and thus the next buffer the server switches to is buffer 2 . Similarly, since $\mathbf{v}_2 . \mathbf{n}_1 \geq 0$ and $\mathbf{v}_3 . \mathbf{n}_2 \geq 0$, the next two buffers visited will be buffers 3 and 1. □
Also it can be shown that

Theorem 2 *If at least one of the inequalities in (3) holds strictly, then the trajectory* $\{\mathbf{x}_n,\ n \geq 1\}$ *converges to a unique periodic orbit with period 3.*

Now we turn to the case where the system has chaotic dynamics. We use two approaches to establish that the conditions given by the theorem ensures chaotic behavior. Approach one is essentially the one used by chase et al. [1].

Theorem 3 *If* $\mathbf{v}_i.\mathbf{n}_j < 0$ *for all* $j \neq i$, $i = 1, 2, 3$ *then the trajectory* $\{\mathbf{x}_n,\ n \geq 1\}$ *is chaotic.*

Before proving this theorem we need some preliminary results. Assume the coordinates of the vertices of the triangle ABC are $(0,0)$, $(b_1, 0)$ and (c_1, c_2), respectively for A, B, and C. Consider the linear transformation Φ on $\mathbf{R}^2$ defined by the matrix

$$A = \frac{1}{6b_1c_2}\begin{pmatrix} 2c_2 & b_1 - 2c_1 \\ 0 & \sqrt{3}b_1 \end{pmatrix}$$

Φ transforms the triangle ABC,corresponding to a corridor policy, into a "standard" equilateral triangle $A'B'C'$ with vertices $(0,0), (1/3, 0)$ and $(1/6, \sqrt{3}/6)$, respectively. Let the inner normals of $A'B'C'$ be $\mathbf{n}'_1$, $\mathbf{n}'_2$, and $\mathbf{n}'_3$ and the velocities be $\mathbf{v}'_1$, $\mathbf{v}'_2$, and $\mathbf{v}'_3$. The following result about the inner product of the velocity and normal vectors of the two triangles follows easily:

Lemma 1 $\mathbf{v}_i.\mathbf{n}_j$ *is positive (negative or zero) if and only if* $\mathbf{v}'_i.\mathbf{n}'_j$ *is positive (negative or zero) for all* $i, j = 1, 2, 3$.

Based on this lemma we can focus on the "standard" triangle and all the results immediately apply to the general triangle through the one-to-one transformation Φ^{-1}. In the following we assume ABC is a "standard" triangle.

Following [1], we define a one-to-one map, ψ, between the triangle and interval $[0, 1)$. Let $x = (x_1, x_2)$ be the coordinates in $\mathbf{R}^2$. Define the map $\psi : \Delta \longrightarrow [0, 1)$ via

$$\psi(x) = \begin{cases} x_1, & \text{if } x \in l_1 \\ \frac{1}{3} + \sqrt{(x_1 - \frac{1}{3})^2 + x_2^2}, & \text{if } x \in l_2 \\ \frac{2}{3} + \sqrt{(x_1 - \frac{1}{6})^2 + (x_2 - \frac{\sqrt{3}}{6})^2}, & \text{if } x \in l_3 - (0,0) \end{cases}$$

Intuitively, the map ψ cuts the triangle ABC at point A(0,0) and unfolds it into the unit interval.

Now let $G : \Delta \longrightarrow \Delta$ be the one step map of the sampled trajectory, i.e., $\mathbf{x}_{n+1} = G(\mathbf{x}_n)$ for all $n \geq 1$. Then $g = \psi G \psi^{-1}$ is a map from unit interval $[0, 1)$ to itself. It is not difficult to calculate that

$$g(x) = \begin{cases} 1 - \frac{2k_1x}{k_1 - \sqrt{3}}, & \text{if } x \in [0, \frac{k_1 - \sqrt{3}}{6k_1}] \\ \frac{1}{3} + \frac{2k_1(1-3x)}{3(k_1 + \sqrt{3})}, & \text{if } x \in [\frac{k_1 - \sqrt{3}}{6k_1}, \frac{1}{3}] \\ \frac{3k_2 + \sqrt{3} - 3(\sqrt{3} + k_2)x}{6k_2}, & \text{if } x \in [\frac{1}{3}, \frac{1}{3} + \frac{2k_2}{3(\sqrt{3} + k_2)}] \\ \frac{2}{3} + \frac{(\sqrt{3} + k_2)(2 - 3x)}{3(\sqrt{3} - k_2)}, & \text{if } x \in [\frac{1}{3} + \frac{2k_2}{3(\sqrt{3} + k_2)}, \frac{2}{3}] \\ \frac{1}{3} + \frac{3\sqrt{3} - k_3 + 3(k_3 - \sqrt{3})x}{3(k_3 + \sqrt{3})}, & \text{if } x \in [\frac{2}{3}, \frac{\sqrt{3} + k_3}{3(\sqrt{3} - k_3)}] \\ \frac{(k_3 - \sqrt{3})(1 - x)}{2k_3}, & \text{if } x \in [\frac{\sqrt{3} + k_3}{3(\sqrt{3} - k_3)}, 1) \end{cases} \tag{4}$$

where k_1, k_2, k_3 is the slope of $\mathbf{v}_1$, $\mathbf{v}_2$ and $\mathbf{v}_3$, respectively.

Note that $g(x)$ is piecewise linear and $|g'(x)| > 1$ for all x (at the points where g is not differentiable both right and left derivatives exist and are greater than one in absolute value). We next show that g is statistically stable. Chaotic dynamics follow. Let $f(x)$ be a probability density function (p.d.f.) on the unit interval. The Frobenius-Perron operator associated with g is defined by $P_g(f)(x) = \frac{d}{dx}\int_{g^{-1}[0,x]} f(s)ds$. After calculating the integral we have

$$P_g(f)(x) = \begin{cases} \frac{2k_2}{\sqrt{3}+k_2} f\left(\frac{3k_2+\sqrt{3}-6k_2x}{3(\sqrt{3}+k_2)}\right) + \frac{2k_3}{k_3-\sqrt{3}} f\left(1-\frac{2k_3x}{k_3-\sqrt{3}}\right), & x \in [0, \frac{1}{3}] \\ \frac{k_1+\sqrt{3}}{2k_1} f\left(\frac{3k_1+\sqrt{3}-3(k_1+\sqrt{3})x}{6k_1}\right) + \frac{\sqrt{3}+k_3}{\sqrt{3}-k_3} f\left(\frac{4\sqrt{3}-3(k_3+\sqrt{3})x}{3(\sqrt{3}-k_3)}\right), & x \in [\frac{1}{3}, \frac{2}{3}] \\ \frac{k_1-\sqrt{3}}{2k_1} f\left(\frac{k_1-\sqrt{3}}{2k_1}(1-x)\right) + \frac{\sqrt{3}-k_2}{\sqrt{3}+k_2} f\left(\frac{4\sqrt{3}-3(\sqrt{3}-k_2)x}{3(\sqrt{3}+k_2)}\right), & x \in [\frac{2}{3}, 1] \end{cases} \quad (5)$$

The stationary density associated with the operator P_g is defined as the limit of $P_g^n(f)$. If there exits a unique p.d.f. f^* such that $P(f^*) = f^*$, and for any p.d.f. f $\lim_{n\to\infty} P^n(f) = f^*$, we say the map g is statistically stable.

We have

Lemma 2 *The map g is statistically stable, and the unique stationary density is the piecewise constant function*

$$f^*(x) = \begin{cases} 2Kk_1(k_3-k_2), & x \in [0, \frac{1}{3}] \\ K(k_2+\sqrt{3})(k_3-k_1), & x \in [\frac{1}{3}, \frac{2}{3}] \\ K(k_3-\sqrt{3})(k_1-k_2), & x \in [\frac{2}{3}, 1] \end{cases} \quad (6)$$

where $K = [k_1(k_3-k_2) + \frac{\sqrt{3}}{3}(k_3 - 2k_1 + k_2)]^{-1}$.

This ends the first approach to establishing chaotic behavior. In the second approach, that extends more easily to the switched server system considered in the next section, we define a one-to-one mapping from the triangle ABC to a circle S in $\mathbf{R}^2$. Let $\Psi : [0,1) \to S$ be defined by $\Psi(x) = exp(2\pi ix)$. Let $h_1 = \Psi^{-1}g\Psi : S \to S$ be the map on the circle S associated with the one step map G on the triangle. These two maps are topologically conjugate (Ψ and ψ are homeomorphisms) hence we can restrict our attention to h_1.

We rely on the following facts:

Theorem (Shub). Let M be a campact C^∞ Riemanian manifold without boundry. Let $h_1, h_2 : M \to M$ be expanding maps and assume that h_1 is homotopic to h_2, then h_1 and h_2 are topologically conjugate [10].

Hopf degree Theorem. Let M be an orientable compact manifold and $h_1, h_2 : M \to S^k$ be C^1 maps from M to the k-sphere S^k. If $deg(h_1) = deg(h_2)$, then h_1 and h_2 are homotopic.

(For the definition of the degree of a map and a reference to the above theorem see, e.g. [6].)

Let $h_2 : S \longrightarrow S$ be defined by $h_2(exp(2\pi ix)) = exp(-2(2\pi ix))$. It is easy to verify that

Lemma 3 *i) h_1 and h_2 are expanding maps.*
ii) $deg(h_1) = deg(h_2) = -2$.

Using modified versions of the above theorems (h_1 and h_2 are piecewise C^∞) we conclude that h_1 and h_2 are homotopic and therefore, topologically conjugate. An analysis similar to the one in [3] for the map $h_3 : S \longrightarrow S$ be defined by $h_3(2\pi ix) = exp(2(2\pi ix))$, shows that h_2 is chaotic and by the above argument so is h_1. □

4 Open Switched Server Systems

In this case $\sum_{i=1}^{3} \rho_i < 1$, i.e., the capacity of server is more than the input workload. Hence the trajectory will eventually converge to the apex of the corridor. As was pointed out we can retain the sequence of buffers visited by the server if we project the trajectory on a plane cutting through the corridor. We consider the plane defined by $x_1+x_2+x_3 = 1$. The projected trajectory on the resulting triangle has a more complex dynamics than the closed switched server system. To each point on a side of the triangle there is an associated velocity vector. These velocity vectors are not parallel to each other, as is the case for a closed system (see Figure 2).

We first "standardize" the problem. Consider a corridor that consists of the planes $x = 0$, $y = 0$ and $z = 0$. It can be verified that any linear corridor can be transformed into this standard corridor and preserve the sign of the inner products of the velocity and normal vectors. In what follows, we shall focus on the standard corridor.

We assume that the following conditions hold:

$$\mathbf{v}_i.\mathbf{n}_j < 0 \quad \text{for } i \neq j \tag{7}$$

Note that the above vectors are vectors in $\mathbf{R}^3$ and not their projections on the triangle. On each side of the triangle there is only one point such that if we start from that point and follow the velocity vector we will hit the opposite axis. We denote those points on the triangle by D_i and those on axis by E_i. The coordinates of D_i can be denoted by $(a_i, 1-a_i, 0)$ and E_i by $(0,0,b_i)$ (see Figure 3). We have the following main result for open switched server systems.

Theorem 4 *If condition (7) holds and $b_i > \max\{a_i, 1-a_i\}$ for $i = 1,2,3$, then the trajectory on triangle is chaotic.*

Proof: We can use a one-to-one mapping similar to ψ, from the triangle to the interval $[0, 3\sqrt{2})$. It can be verified that the one step map of the triangle translates into the following map on the $[0, 3\sqrt{2})$ interval.

$$G(x) = \begin{cases} 2\sqrt{2} + \frac{\sqrt{2}(1-a_1)-x}{1-a_1-x(1-b_1)/\sqrt{2}}, & \text{if } x \in [0, \sqrt{2}(1-a_1)] \\ \sqrt{2} + \frac{\sqrt{2}b_1(\sqrt{2}-x)}{\sqrt{2}(a_1+b_1-1)+x(1-b_1)}, & \text{if } x \in [\sqrt{2}(1-a_1), \sqrt{2}] \\ \frac{\sqrt{2}(1-a_2)-(x-\sqrt{2})}{1-a_2-(x-\sqrt{2})(1-b_2)/\sqrt{2}}, & \text{if } x \in [\sqrt{2}, \sqrt{2}(2-a_2)] \\ 2\sqrt{2} + \frac{\sqrt{2}b_2(2\sqrt{2}-x)}{\sqrt{2}(a_2+b_2-1)+(x-\sqrt{2})(1-b_2)}, & \text{if } x \in [\sqrt{2}(2-a_2), 2\sqrt{2}] \\ \sqrt{2} + \frac{\sqrt{2}(1-a_3)-(x-2\sqrt{2})}{1-a_3-(x-2\sqrt{2})(1-b_3)/\sqrt{2}}, & \text{if } x \in [2\sqrt{2}, \sqrt{2}(3-a_3)] \\ \frac{\sqrt{2}b_3(3\sqrt{2}-x)}{\sqrt{2}(a_3+b_3-1)+(x-2\sqrt{2})(1-b_3)}, & \text{if } x \in [\sqrt{2}(3-a_3), 3\sqrt{2}] \end{cases} \tag{8}$$

It can be checked that under the conditions of the thoerem this map is expanding. Moreover, once this map is transformed to a map from S to S it can be easily verified that the degree of the map is equal to -2. Hence it is topologically conjugate to the map h_2 defined in the previous section and is chaotic.

5 Conclusions

Two classes of switched server systems are considered under corridor policies. We have investigated the influence of the parameters of the corridor on the dynamics of the buffer states. We have derived conditions on the parameters of the corridor under which the system behaves periodically and also those under which the dynamics of the system is chaotic. In all cases we have assumed the set-up times (switch-over times) are zero. Studying the behavior of these systems under non-zer set-up times is worthwhile. Also we have not considered an intermediate range between the two conditions given in the paper. This is a future project.

References

[1] C. Chase, J. Serrano, and P. Ramadge. Periodicity and chaos from switched flow systems: Contrasting examples of discretely controlled continuous systems. *IEEE Trans. on Automatic Control*, 38(1), pp. 70–83, Jan. 1993.

[2] H. Chen and D. Yao. A fluid model for systems with disturbances. *Operations Research*, 40, pp. 239–247, May-June 1992.

[3] R.L. Devaney. *Chaotic Dynamical Systems.* Addison-Wesley, New York, NY, 1989.

[4] C. Horn, and P. Ramadge. Dynamics of Switched Arrival Systems with Thresholds. manuscript, 1993.

[5] S.B. Gershwin. Hierarchical Flow Control: A Framework for Planning and Scheduling Discrete Events in Manufacturing Systems. *Proceedings of IEEE*, Vol. 77, No. 1, pp. 195-209, Jan. 1989

[6] V. Guillemin and A. Pollack. *Differential Topology* Prentice Hall, 1974.

[7] J.R. Perkins and P.R. Kumar. Stable, distributed, real-time, scheduling of flexible manufacturing/assembly/disassembly systems. *IEEE Trans. on Automatic Control*, 34(2), pp. 139–148, Feb. 1989.

[8] A. Sharifnia, M. Caramanis, and S. B. Gershwin. Dynamic set-up scheduling and flow control in manufacturing systems. *Discrete Event Dynamic Systems*, pp. 149–175, 1991.

[9] A. Sharifnia. Stability and Performance of Distributed Production Control Methods based on Continuous Flow Models. to appear in *IEEE Trans. on Automatic Control.*

[10] M. Shub. Expanding Maps. *Global Analysis, Proceedings of Symposia in Pure Mathematics*, Vol. XIV, pp. 273-276, 1968.

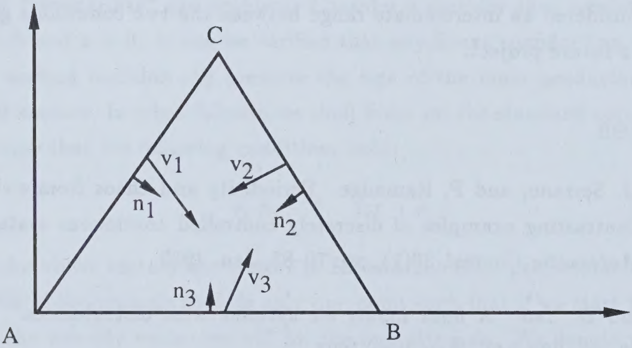


Figure 1: Closed Switched Server System

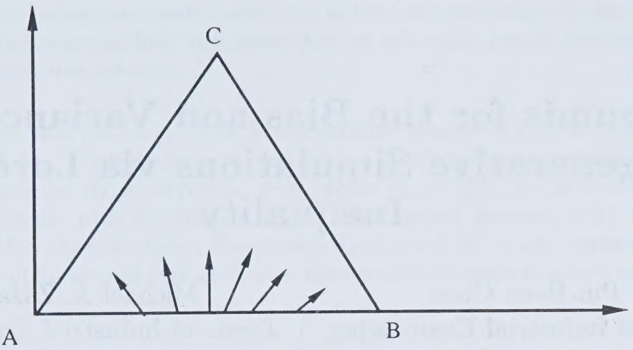


Figure 2: Open Switched Server System (velocity of one side is projected)

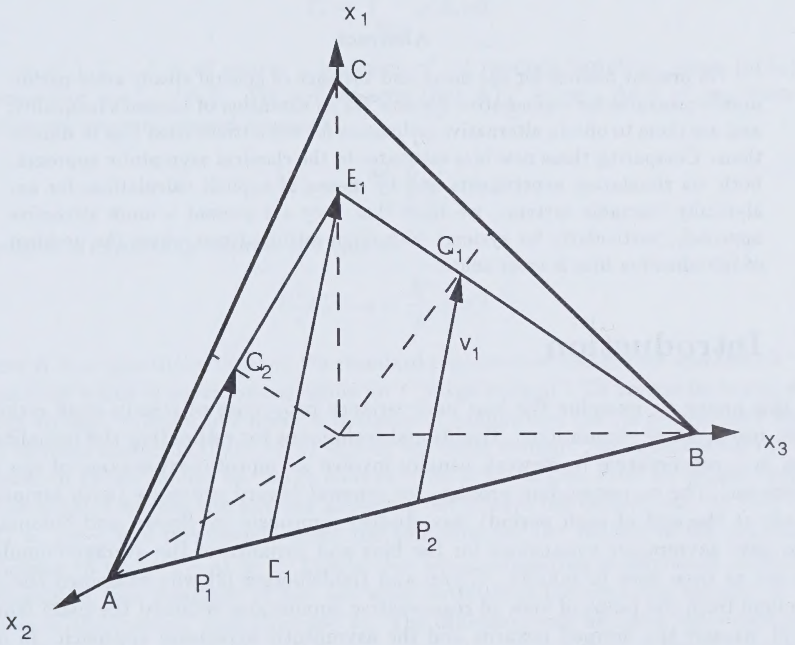

x1
C
E1
C1
v1
C2
B
x3
P2
E1
P1
A
x2

Figure 3: Open Switched Server System

Bounds for the Bias and Variance in Regenerative Simulations via Lorden's Inequality

Pin-Hsun Chen
Dept. of Industrial Engineering
Northwestern University
Evanston, IL 60208

Michael A. Zazanis
Dept. of Industrial Engineering
University of Massachusetts
Amherst, MA 01003

Abstract

We present bounds for the mean and variance of general steady state performance measures for regenerative systems via an extension of Lorden's inequality, and use them to obtain alternative estimators for the initialization bias in simulations. Comparing these new bias estimates to the classical asymptotic approach, both via simulation experiments and by means of explicit calculations for analytically tractable systems, we show that they are present a more attractive approach, particularly for systems with large settling times where the problem of initialization bias is most acute.

1 Introduction

In this paper we examine the bias and variance properties of steady state estimates from regenerative simulations. Traditional techniques for estimating the initialization bias in a regenerative framework usually involve asymptotic expansions of the ratio estimator. The corresponding problem for renewal-reward processes (with lumped rewards at the end of each period) was studied rigorously by Brown and Solomon [2] who gave asymptotic expansions for the bias and variance of the average cumulative reward as time goes to infinity. Glynn and Heidelberger [3] who examined the same problem from the point of view of regenerative simulations retained the main features of [2], namely the lumped rewards and the asymptotic expansion approach. In many instances however simulation problems give rise to renewal-reward processes with a continuous reward rate (as for example when one is estimating the percentage of time a machine is idle or the inventory is above a certain level). For this reason we felt that continuous reward rates are often a more natural assumption in a simulation context.

In a far more important departure from the standard approaches to the problem, we abandon asymptotic expansions altogether in favor of bounds which hold for all t

and which we believe are more useful than asymptotic expansions in determining the size of the initialization bias in a sense that we will make precise presently. We will first introduce some notation.

Assume that $\{Z_s; s \geq 0\}$ is a regenerative process w.r.t. a renewal process $\{S_n;\ n = 0, 1, \ldots\}$ where $S_0 = 0$, and $\{N_t;\ t \geq 0\}$ is the corresponding counting process defined as $N_t = \inf\{n;\ S_n > t\}$ where $S_n = \sum_{i=1}^{n} X_i$, $n = 1, 2, \ldots$. $\{X_i\}$ is an iid sequence with distribution F and finite second moment, $EX_1^2 < \infty$. With $g : \mathbf{R} \to \mathbf{R}^+$ a Borel function, the process $\{g(Z_s); s \geq 0\}$ is also regenerative w.r.t. $\{S_n\}$. With $g(Z_s)$ reward rate at time s, the cumulative reward up to t is

$$C_t = \int_0^t g(Z_s)ds,$$

and the reward accumulated during the n-th renewal interval is

$$Y_n = \int_{S_{n-1}}^{S_n} g(Z_t)dt.$$

$\{Y_n\ ; n = 1, 2, \ldots\}$ is of course a sequence of iid random variables, since $\{g(Z_s)\}$ is regenerative w.r.t. $\{S_n\}$. We will assume that $EY_1^2 < \infty$. As $t \to \infty$, from the elementary renewal-reward theorem,

$$\frac{1}{t}C_t \to a \overset{\text{def}}{=} \frac{EY_1}{EX_1} \quad \text{w.p.1.}$$

Asymptotic expansions express the bias as

$$E\frac{1}{t}C_t - a = \frac{K}{t} + o(t^{-1}), \tag{1}$$

where K is a quantity estimated via standard regenerative techniques and the $o(t^{-1})$ is a quantity which is assumed negligible for t "large enough". Of course to decide what "large enough" means, one needs a measure of comparison (i.e., a proper time scale). The obvious answer is that t/EX_1 should be "large enough", or equivalently that the number of cycles should be "large enough". As it turns out, even for simple systems as the M/M/1 queues the obvious answer is not always the correct one. In that case, $EX_1 = \frac{1}{\lambda(1-\rho)}$ and one would expect the condition for the validity of the asymptotic expansion to be $\lambda t(1-\rho) \gg 1$. This is true when estimating steady state occupation probabilities (see Example 1) but *not* when estimating the expected number of customers in the system. (see Example 2). The difference can be quite significant when utilization is high. Obviously t/EX_1 is not enough to ensure the validity of the asymptotic expansion and in general the proper time scale will be unknown. The bounds we propose on the other hand are valid for all t and provide a more reliable indicator of when one can safely ignore initialization bias.

Section 2 contains the derivation of the bounds for the bias of $\frac{1}{t}C_t$ based on Lorden's inequality whereas §3 derives an upper bound for its variance. In section 4 a new asymptotic expansion is obtained for the continuous time reward case, several examples of analytically tractable systems and explicit expressions for the bias of steady state estimators are given.

2 Bounds for the Expectation of $\frac{1}{t}C_t$

To obtain bounds for the bias $E[\frac{1}{t}C_t - a]$ we will use inequalities for the renewal function derived by Lorden [4] (see also Asmussen [1, 158-159]). Let $R_t = S_{N_t} - t$ and $V_t = t - S_{N_t - 1}$ be the excess and age of the renewal process $\{S_n\}$ at t respectively. In [4] elegant and elementary subadditivity arguments are used to prove the following inequalities:

$$ER_t \leq \frac{E[X_1^2]}{EX_1}, \tag{2}$$

$$EV_t \leq \frac{E[X_1^2]}{EX_1}, \tag{3}$$

$$\text{and} \quad \frac{t}{EX_1} \leq EN_t \leq \frac{t}{EX_1} + \frac{E[X_1^2]}{(EX_1)^2}. \tag{4}$$

Letting

$$\frac{E[X_1^2]}{(EX_1)^2} = b,$$

we can state Lorden's bound for renewal function as follows:

$$\frac{t}{EX_1} \leq EN_t \leq \frac{t}{EX_1} + b. \tag{5}$$

The above inequality gives the tightest bounds available involving only the first two moments of the interarrival times X_i and making no assumption about their distribution, F. Under the assumption that F is DFR or IMRL a tighter bound is possible. (See Brown [2]. For a brief survey of available bounds for the renewal function see Stoyan [5, 130-133].) In the context of regenerative simulation it is usually impossible to verify whether F is DFR or not and thus (5) is ideal for our purpose.

Using (5) and Wald's lemma bounds for the bias in a system with lumped rewards can be obtained readily. Let $D_t = \sum_1^{N_t} Y_i$ be the total reward accumulated by the end of the renewal interval containing t. Then

$$\frac{t\,EY_1}{EX_1} \leq ED_t = EN_t EY_1 \leq (\frac{t}{EX_1} + b)EY_1$$

or equivalently

$$0 \leq \frac{1}{t}ED_t - \frac{EY_1}{EX_1} \leq \frac{b}{t}EY_1. \tag{6}$$

We now turn our attention to the continuous renewal-reward process. The main idea behind our analysis involves a random time change with clock rate equal to the reward rate. Formally, the random time $\tau(t)$ corresponding to t is

$$\tau(t) = \int_0^t g(Z_s)ds,$$

i.e. equal to the cumulative reward C_t and, conversely,

$$t(\tau) = \inf\{u : \int_0^u g(Z_s)ds \geq \tau\}.$$

Under this time change the original renewal process $\{S_n; n = 0, 1, \ldots\}$ is transformed into a point process $\{\Phi_n; n = 0, 1, \ldots\}$ which is also renewal since for all $n = 1, 2, \ldots$, $\Phi_n - \Phi_{n-1} = \int_{S_{n-1}}^{S_n} g(Z_s)ds = Y_n$, which are iid (see Figure 1). In the figure, the black and hollow circles refer to the renewal points of $\{S_n\}$ and $\{\Phi_n\}$ respectively.

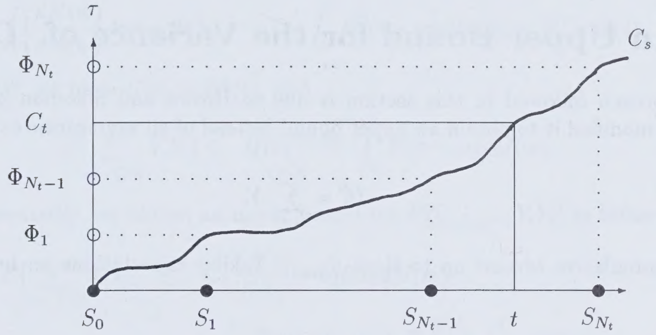


Figure 1.

We can now apply Lorden's inequality to the time-transformed renewal process. It is clear that $\Phi_{N_t} - C_t$ is the excess at C_t of the renewal process $\{\Phi_n\}$. Using (2) once more, we obtain

$$E[\Phi_{N_t} - C_t] \leq \frac{E[(\Phi_{S_1} - \Phi_{S_0})^2]}{E[\Phi_{S_1} - \Phi_{S_0}]},$$

or equivalently

$$E[\int_t^{S_{N_t}} g(Z_s)ds] \leq \frac{E[Y_1^2]}{EY_1}. \tag{7}$$

Now we can acquire a lower bound for EC_t as follows:

$$EC_t = E[\int_0^t g(Z_s)ds] = E[D_t - \int_t^{S_{N_t}} g(Z_s)ds].$$

Therefore

$$EC_t \geq EN_tEY_1 - \frac{E[Y_1^2]}{EY_1} \geq \frac{t\ EY_1}{EX_1} - \frac{E[Y_1^2]}{EY_1}. \tag{8}$$

For small values of t, the right-hand side of (8) becomes negative and the lower bound for EC_t becomes trivial since g (and therefore EC_t) is assumed positive. Thus

$$\max\{0, \frac{t\ EY_1}{EX_1} - \frac{EY_1^2}{EY_1}\} \leq EC_t \leq ED_t \leq (\frac{t}{EX_1} + b)EY_1, \tag{9}$$

Denoting by $B(t) = E[\frac{1}{t}C_t] - \frac{EY_1}{EX_1}$, the bias of the estimator $\frac{1}{t}C_t$ satisfies

$$-\min\left\{\frac{EY_1}{EX_1}, \frac{EY_1^2}{tEY_1}\right\} \leq B(t) \leq \frac{b}{t}EY_1. \tag{10}$$

3 An Upper Bound for the Variance of $\frac{1}{t}C_t$

The approach followed in this section is due to Brown and Solomon [2]. We have suitably modified it to obtain an upper bound instead of an asymptotic expansion. Let

$$D_t^0 = \sum_{i=1}^{N_t-1} Y_i$$

be the cumulative reward up to time S_{N_t-1}. Taking expectations on both sides, we have

$$\begin{aligned} ED_t^0 &= E[\sum_{i=1}^{\infty} Y_i 1_{(S_i \leq t)}] = \sum_{i=1}^{\infty} E\{E[Y_i 1_{(S_i \leq t)} \mid S_i]\} \\ &= \sum_{i=1}^{\infty} \int_0^t E(Y_i | S_i = s) dF^i(s), \end{aligned}$$

where F is the distribution function of X_i and F^i is the i-fold convolution of F with itself (i.e. F^i is the distribution function of $S_i = \sum_{n=1}^{i} X_n$). If $m(s) = \sum_{i=0}^{\infty} F^i(s)$ is the renewal function corresponding to the distribution F, then it is clear that for all i, F^i is dominated by m ($m \gg F^i$) since that for all Borel sets B such that $m(B) = 0$, $F^i(B)$ is equal to 0. From the Radon-Nikodym theorem it follows that there exists a “density” function $f_i(s)$ such that $dF^i(s) = f_i(s)dm(s)$. If we denote by $H^0(t)$ the expected cumulative reward ED_t^0, then

$$H^0(t) = \sum_{i=1}^{\infty} \int_0^t E(Y_i | S_i = s) f_i(s) dm(s).$$

Similarly let $H(t) = ED_t$, where $D_t = \sum_{i=1}^{N_t} Y_i$ is the cumulative reward up to time S_{N_t}. Then

$$H(t) = E[\sum_{i=1}^{N_t} Y_i] = E[\sum_{i=1}^{\infty} Y_i 1_{(S_{i-1} \leq t)}] = \int_0^t \sum_{i=1}^{\infty} E[Y_i | S_{i-1} = s] f_{i-1}(s) dm(s),$$

and

$$\frac{dH(t)}{dm(t)} = \sum_{i=1}^{\infty} E(Y_i | S_{i-1} = s) f_{i-1}(s).$$

Thus

$$\begin{aligned} E[\sum_{i<j\leq N_t} Y_i Y_j] &= E[\sum_{i<j} Y_i Y_j 1_{(S_{j-1} \leq t)}] \\ &= \sum_{j=1}^{\infty} \sum_{i=1}^{j-1} \int_0^t E[Y_i | S_{i-1} = w] dF^{i-1}(w) \int_w^t E[Y_j | S_{j-1} = s] dF^{j-1}(s) \\ &= \int_0^t \int_{v=0}^{v=t-w} \sum_{i=1}^{\infty} E[Y_i | S_{i-1} = w] f_{i-1}(w) \sum_{k=1}^{\infty} E[Y_k | S_k = v] dF^k(v) dm(w) \\ &= \int_0^t \frac{dH(w)}{dm(w)} dm(w) H^0(t-w) = \int_0^t H^0(t-w) dH(w) = H^0 * H(t). \end{aligned}$$

Since $D_t \geq D_t^0$, we have $H(t) \geq H^0(t)$, and

$$E[\sum_{i<j\leq N_t} Y_i Y_j] \leq \; H(t)^{*2} \overset{\text{def}}{=} \int_0^t H(t-w) dH(w).$$

Using (6) repeatedly, we obtain an upper bound for $E[\sum_{i<j\leq N_t} Y_i Y_j]$ as follows:

$$\begin{aligned} E[\sum_{i<j\leq N_t} Y_i Y_j] &\leq \int_0^t (\frac{t-w}{EX_1} + b) EY_1 dH(w) \\ &= (\frac{t}{EX_1} + b) EY_1 H(t) - \frac{EY_1}{EX_1} \int_0^t w dH(w) \\ &= bEY_1 H(t) + \lambda EY_1 \int_0^t H(w) dw \\ &\leq bEY_1 H(t) + \frac{EY_1}{EX_1} \int_0^t (\frac{w}{EX_1} + b) EY_1 dw \\ &\leq b(EY_1)^2 (\frac{t}{EX_1} + b) + \frac{(EY_1)^2}{EX_1} (\frac{t^2}{2EX_1} + bt) \\ &= \frac{(EY_1)^2}{2} \left\{ \frac{t^2}{(EX_1)^2} + \frac{4tb}{EX_1} + 2b^2 \right\}. \end{aligned} \tag{11}$$

Now we can find an upper bound for the variance of C_t, $Var[C_t]$, using the inequality $C_t \leq D_t$:

$$Var[C_t] \; = \; E[C_t^2] - \{EC_t\}^2 \leq E[D_t^2] - \{EC_t\}^2 = E[(\sum_{i=1}^{N_t} Y_i)^2] - \{EC_t\}^2$$

$$= E\sum_{i=1}^{N_t} Y_i^2 + 2E[\sum_{i<j\leq N_t} Y_iY_j] - \{EC_t\}^2$$

$$= EN_tEY_1^2 + 2E[\sum_{i<j\leq N_t} Y_iY_j] - \{EC_t\}^2. \tag{12}$$

From (5) and (11), we have

$$Var[C_t] \leq (\frac{t}{EX_1} + b)EY_1^2 + (EY_1)^2\{\frac{t^2}{(EX_1)^2} + \frac{4tb}{EX_1} + 2b^2\} - \{EC_t\}^2. \tag{13}$$

Therefore, an upper bound for $Var[\frac{1}{t}C_t]$ can be obtained for which we distinguish two cases:

(I) For $t \geq \frac{EX_1EY_1^2}{(EY_1)^2}$

$$\begin{aligned} Var[\frac{1}{t}C_t] \leq\ & \frac{1}{t^2}\Bigg[(\frac{t}{EX_1} + b)EY^2 + (EY_1)^2(\frac{t^2}{E[X_1^2]} + \frac{4tb}{EX_1} + 2b^2) - \frac{(EY_1)^2t^2}{(EX_1)^2} + \\ & \frac{2tEY_1^2}{EX_1} - (\frac{EY_1^2}{EY_1})^2\Bigg] \\ =\ & \frac{1}{t}\left[\frac{4b(EY_1)^2}{EX_1} + 3\frac{EY_1^2}{EX_1}\right] + \frac{1}{t^2}\left[bEY_1^2 + 2b^2(EY_1)^2 - (\frac{EY_1^2}{EY_1})^2\right]. \end{aligned}$$

(II) For $t < \frac{EX_1EY_1^2}{(EY_1)^2}$

$$Var[\frac{1}{t}C_t] \leq \frac{1}{t^2}\left[(\frac{t}{EX_1} + b)EY^2 + (EY_1)^2(\frac{t^2}{E[X_1^2]} + \frac{4tb}{EX_1} + 2b^2)\right].$$

4 Applications to Regenerative Simulations

Renewal-reward processes with lumped rewards Y_i occurring at the end of each cycle have been used to study the bias properties of regenerative simulations in [3]. The corresponding asymptotic expansion is

$$E[\frac{1}{t}\sum_{i=1}^{N_t-1} Y_i] = \frac{EY_1}{EX_1} + \frac{D_1}{t} + o(t^{-1}),$$

where $D_1 = \frac{EX_1^2EY_1}{2(EX_1)^2} - \frac{E[X_1Y_1]}{EX_1}$. A new asymptotic expansion is obtained in this section for the continuous time reward case. Indeed we will show that:

$$E[\frac{1}{t}C_t] = E[\frac{1}{t}\sum_{i=1}^{N_t-1} Y_i] + E[\frac{1}{t}\int_{S_{N_t-1}}^{t} g(Z_s)ds] = \frac{EY_1}{EX_1} + \frac{D_2}{t} + o(t^{-1}), \tag{14}$$

where $D_2 = \frac{EY_1 E[X_1^2]}{2(EX_1)^2} - \frac{\int_0^{X_1} sg(Z_s)ds}{EX_1}$. The derivation of D_2 is based on the following renewal theoretic argument. Let $G(t) = EC_t = E\int_0^t g(Z_s)ds$. Then

$$\begin{aligned} G(t) &= E[C_t; t \geq S_1] + E[C_t; t < S_1] \\ &= \int_0^t G(t-s)dF(s) + \int_0^t E[Y_1|X_1 = s]dF(s) + E[C_t; t < X_1], \end{aligned}$$

where $Y_1 = \int_0^{X_1} g(Z_s)ds$. $G(t) - at$ satisfies the following renewal equation:

$$\begin{aligned} G(t) - at &= \int_0^t [G(t-s) - a\,(t-s)]dF(s) + \int_0^t a\,(t-s)dF(s) - at \\ &\quad + \int_0^t E[Y_1|X_1 = s]dF(s) \ + \ E[C_t; t < X_1] \\ &= \int_0^t [G(t-s) - a(t-s)]dF(s) \\ &\quad + a\int_t^\infty [1 - F(s)]ds - \int_t^\infty E[Y_1|X_1 = s]dF(s) + E[C_t; t < X_1]. \end{aligned}$$

Since $E[C_t;\ t < X_1]$ is directly Riemann integerable an appeal to the key renewal theorem yields

$$\begin{aligned} \lim_{t\to\infty}[G(t) - at] &= \frac{1}{EX_1}\int_0^\infty \Big\{ a\int_t^\infty [1 - F(s)]ds \\ &\qquad - \int_t^\infty E[Y_1|X_1 = s]dF(s) + E[C_t; t < X_1]\Big\}\,dt \\ &= D_1 + \frac{1}{EX_1}\int_0^\infty E[C_t; t < X_1]dt, \end{aligned}$$

with $D_1 = \frac{EX_1^2 EY_1}{2(EX_1)^2} - \frac{E[X_1Y_1]}{EX_1}$. The second term on the right-hand-side of the above equation can be rewritten as

$$\begin{aligned} \int_0^\infty E[C_t; t < X_1]dt &= E[\int_0^{X_1} C_t dt] = E[\int_0^{X_1}\int_0^t g(Z_s)dsdt] \\ &= E[\int_0^{X_1} g(Z_s)(X_1 - s)ds] = E[X_1Y_1] - E[\int_0^{X_1} sg(Z_s)ds]. \end{aligned}$$

Therefore,

$$\begin{aligned} \lim_{t\to\infty}[G(t) - at] &= D_1 + \frac{1}{EX_1}\Big\{E[X_1Y_1] - E[\int_0^{X_1} sg(Z_s)ds]\Big\} \\ &= \frac{EY_1 E[X_1^2]}{2(EX_1)^2} - \frac{E\int_0^{X_1} sg(Z_s)ds}{EX_1}. \end{aligned}$$

Consequently,

$$E[\frac{1}{t}C_t] = \frac{EY_1}{EX_1} + \frac{1}{t}\left\{\frac{EY_1 E[X_1^2]}{2(EX_1)^2} - \frac{E\int_0^{X_1} sg(Z_s)ds}{EX_1}\right\} + o(1/t).$$

Typically, D_1 and D_2 are estimated from the data and for large t, higher order terms are discarded. While often appropriate, this method has drawbacks that become clear if we consider a family of systems depending on a parameter. For the sake of concreteness consider the following examples.

Example 1 Consider a family of $M/M/1$ queueing processes parametrized by the utilization $\rho \in (0,1)$. Let X_1 be the length of the first busy cycle of the system and $Y_1 = \int_0^{X_1} 1_{(N_s \geq 2)} ds$ the amount of time that there are more than two customers in the system during the first busy cycle, and $C_t = \int_0^t 1_{(N_s \geq 2)} ds$ the amount of time that there are more than two customers in the system in $[0,t]$. Then the estimator $\frac{1}{t}C_t$ of the steady state probability $P(N \geq 2)$ has bias $B(t)$, given by

$$B(t) = E[\frac{1}{t}C_t] - \frac{EY_1}{EX_1} = \frac{\rho^2(1-3\rho)}{(1-\rho)\lambda t} + o(\frac{1}{(1-\rho)\lambda t}), \tag{15}$$

where λ is the arrival rate. From the above expression, it becomes clear that if the asymptotic expansion is used in heavy traffic (i.e. $\rho \to 1$) the appropriate time scale is given by the change of variables $\tau = (1-\rho)\lambda t$. Then the bias (15) can be written as $\frac{\rho^2(1-3\rho)}{\tau} + o(\tau^{-1})$, where $o(\tau^{-1})$ is *uniformly* small as $\tau \to \infty$ for all $\rho \in (0,1)$.

Example 2 Consider next a family of M/M/1 queues with utilization $\rho \in (0,1)$, and lumped reward process $\{W_n\}$, where W_n is the system time of the n-th customer. Let X_1 be the length of the first cycle and Q_1 the number of arrivals in that cycle. Define $Y_1 = \sum_{i=1}^{Q_1} W_i$. Then the estimator of the average number of customers in the system in steady state, $\frac{1}{t}D_t = \frac{1}{t}\sum_{i=1}^{N_t-1} Y_i$, and

$$B(t) = E[\frac{1}{t}D_t] - \frac{EY_1}{EX_1} = \frac{3\rho^4 - 3\rho^2 + \rho}{\lambda t(1-\rho)^3} + o(\frac{1}{\lambda t(1-\rho)^3}).$$

It is clear that here the appropriate time scale is $\lambda t(1-\rho)^3$.

Example 3 Finally consider a closed queueing network composed of two servers in tandem and N customers, where customers finishing service at one server join the queue at the other. Let λ and μ be the service rates of the first server and the second respectively, and Z_t the number of customers at the second station (including the one being servered) at time t. Considering the number of customers at the second station as a state, let Y_1 be the first passage time from state N to state $N-1$ and X_1 be the first passage time from state N back to state N. Define $C_t = \int_0^t 1_{(Z_s = N)} ds$, then the bias $B(t)$ of estimator $\frac{1}{t}C_t$ of the steady state probability $P(Z = N)$ is

$$B(t) = E[\frac{1}{t}C_t] - \frac{EY_1}{EX_1} = \frac{D(\rho, N)}{\mu t(1-\rho^{N+1})^2} + o(\frac{1}{\mu t(1-\rho^{N+1})^2}),$$

where $\rho = \frac{\lambda}{\mu}$ and $D(\rho, N) = \rho - (2N+1)\rho^N + (2N+3)\rho^{N+1} + 2\rho^{N+2} - \rho^{2N+1} - \rho^{2N+2} + \rho^{2N+3}$.

The above examples indicate that when asymptotic expansions are used to estimate the size of initialization bias, "t" is not always an appropriate time scale. While for simple markovian systems such as the above examples, the appropriate time scale can be obtained with little effort, this may not be possible for general regenerative systems. When this is the case, we can not be sure about the validity of asymptotic expansions for the bias. On the other hand, since the bounds obtained in section 2 hold for all t, we can use them without having to decide whether t is "large" or not (i.e., without having to determine the appropriate time scaling factor).

References

[1] Asmussen S. (1987). *Applied Probability and Queues*, John Wiley & Sons.

[2] Brown, M. and H. Solomon (1975). "A second-order approximation for the variance of a renewal reward process", *Stochastic Processes and Their Applications*, 3, 301-314.

[3] P. W. Glynn and Heidelberger P. (1990). "Bias Properties of Budget constrained Simulations",*Opns. Res.* 38, No. 5, 801-814.

[4] Lorden, G. (1970). "On excess over the Boundary", *Annals of Math. Stat.*, Vol. 41, No. 2, 520-527.

[5] Stoyan, D. (1983). *Comparison Methods for Queues and Other Stochastic Models.* Wiley, New York.

On the computation of normal languages[1]

M. Barbeau, G. Custeau, and R. St-Denis

Département de mathématiques et d'informatique
Université de Sherbrooke
Sherbrooke (Québec)
CANADA J1K 2R1

Abstract

This paper addresses the problem of computing the largest M-normal sublanguage of a language L with respect to a language L(G). M is called an observation function and renames strings of events of L and L(G). The largest M-normal sublanguage is defined as the set of strings s such that s is in L and whenever there is a string s' in L(G) such that M(s')=M(s), s' is in L as well. This problem is related to the supervisory control problem in which observation of events is assumed to be imperfect.

This paper surveys constructive solutions to this problem, including one from the current authors. Five algorithms are introduced and compared with respect to their restrictions, number of intermediate automata, and ease of understanding.

1. Introduction

This paper addresses the problem of computing L_R, the largest M-normal sublanguage of a language L with respect to a language L(G). Automaton G models the behavior of a plant, whereas L represents the control constraints. M is called an observation function and renames strings of events of L and L(G). Language L_R is defined as the set of strings s such that s is in L and whenever there is a string s' in L(G) which is observed as s (i.e., M(s)=M(s')), s' is in L as well. In [1], this language is called the largest closed (M,L(G))-recognizable sublanguage of L. This problem is related to the supervisory control problem in which observation of events is assumed to be imperfect [2,3]. The intuitive interpretation of language L_R is that indistinguishable plant behaviors are all permitted by the controller if they are all legal, or all rejected if at least one of them is illegal.

This paper surveys constructive solutions to the problem of computing L_R. Five algorithms are introduced and compared with respect to their restrictions, number of intermediate automata, and ease of understanding. Theoretically, the worst-case time complexity is exponential with

---

[1] The research described in this paper was supported in part by the Natural Sciences and Engineering Research Council of Canada (NSERC) and the Fonds pour la formation de chercheurs et l'aide à la recherche (FCAR).

all the algorithms. However, in practice, time complexity is less than exponential and most of the real problems are tractable.

The paper is organized as follows. Section 2 provides some preliminary definitions. Section 3 introduces the algorithms for computing the largest M-normal sublanguage of a given language. Section 4 gives a comparison of the algorithms followed by the conclusion.

2. Preliminaries

Let Σ denote a finite set of events. The set of all strings of events is denoted as Σ^*. The empty string is represented as ε. A subset $L \subseteq \Sigma^*$ is called a language over Σ. The prefix closure of L is denoted as $\overline{L}$ and is the set of all strings that are prefixes of strings in L. If $L=\overline{L}$, then L is said to be *prefix closed.* $\overline{s}$ denotes the set of strings that are prefixes of a string s.

Let Λ be another finite set of events. The observation function $M:\Sigma \rightarrow \Lambda \cup \{\varepsilon\}$ represents partial observation by a controller. If $M(\sigma)=\varepsilon$, we say that σ is unobservable and if $M(\sigma)=M(\sigma')$, we say that σ and σ' are indistinguishable.

A deterministic finite automaton (or generator) is a 5-tuple $G=(Q,\Sigma,\delta,q_0,Q_m)$, where Q is a finite set of states, $q_0 \in Q$ is the initial state, $Q_m \subseteq Q$ is the set of marked states, and $\delta:\Sigma \times Q \rightarrow Q$ is the transition function (as usual, δ is extended to $\Sigma^* \times Q$). We denote as L(G) the language generated by G, and $L_m(G)$ the language marked by G, which are defined as:

$$L(G) = \{ s \in \Sigma^* \mid \delta(s,q_0)! \}$$

$$L_m(G) = \{ s \in \Sigma^* \mid (\exists q \in Q_m)[q=\delta(s,q_0)!] \}$$

3. Presentation of the algorithms

All the algorithms accept as input two deterministic automata, G and $K=(X,\Sigma,\xi,x_0,X_m)$, that model the plant behavior and the legal language L, respectively. Without loss of generality, we assume that $L \subseteq L(G)$.

Algorithm 1. To compute an automaton that generates the largest M-normal sublanguage L_R of a language L with respect to L(G), Cieslak et al. [1] proceed as follows.

1. Construction of ξ_R and δ_R, two intermediate transition functions. The transition function ξ_R is constructed to identify the states of K that the controller cannot distinguish as it observes the behavior of the plant through the observation function M. Similarly, the transition function δ_R is constructed to identify the states of the plant that the controller cannot distinguish.

2. Construction of the automaton $\hat{T}$ such that $L(\hat{T})=L_R$, from the transition functions ξ and ξ_R. The transition function $\hat{g}$ of $\hat{T}$ determines how the controller observes the legal behavior of

the plant. If, for a given transition, the set of indistinguishable states obtained in δ_R is different from the one obtained in $\hat{g}$, then there exists an illegal transition that is observed as a legal transition by the controller. This legal transition must therefore be removed. The result of this process is L_R.

The algorithm works under the assumption that $L_m(K)$ and $L_m(G)$ are closed. We found this algorithm particularly difficult to understand. The reader is referred to [1] for the details.

Algorithm 2. The second algorithm comes from [4] and is based on the following formula: $L_R=L-M^{-1}M(L(G)-L)$ [5]. This formula can be interpreted as follows. Take the set of illegal strings, that is, $L(G)-L$. Compute how the illegal strings are observed, that is, $M(L(G)-L)$. Compute the set of strings, legal or illegal, that are observed as illegal strings, that is, $M^{-1}M(L(G)-L)$. Finally, reject from L the legal strings for which there are illegal strings observed the same way.

Algorithm 2 restricts the observation function M to natural projections, that is, $M(\sigma)=\sigma$ or $M(\sigma)=\varepsilon$.

This formula can be effectively computed with the TCT tool [4, 6]. The algorithm results from the breakdown of the above formula into a series of unary or binary operations. The process is as follows:

1. Compute an automaton D_2 that generates the set of illegal strings of G, that is, the language $L(G)-L$. Since TCT has no difference operation, this is performed in two steps. First, compute an automaton D_1 that generates Σ^*-L, the complement of L. Second, compute D_2, an automaton that generates $L(G)\cap L(D_1)$. The intersection of two languages involves taking the Cartesian product of states that belong to G and D_1, respectively.

2. Compute a nondeterministic automaton D_3 that generates $M(L(D_2))$. This is done by renaming the events in D_2 according to M.

3. Calculate an automaton D_4 that generates $M^{-1}(L(D_3))$. This is done by adding to every state of D_3 a selfloop on every nonobservable event.

4. Construct, in two steps, an automaton D_6 that generates $L-L(D_4)$. First, compute an automaton D_5 that generates $\Sigma^*-L(D_4)$. Second, calculate an automaton D_6 that generates $L\cap L(D_5)$. Language $L(D_6)$ is equal to L_R.

Algorithm 3. The next algorithm is introduced in [7]. Its presentation needs two preliminary definitions. Let K and G be as defined above. We say that K is a *subautomaton* of G if the following conditions are satisfied: i) $X\subseteq Q$, $x_0=q_0$, and ii) for all $x\in X$, $\sigma\in\Sigma$, if $\xi(\sigma,x)!$

then $\xi(\sigma,x)=\delta(\sigma,x)$. K is a *strict-subautomaton* of G if the following conditions are satisfied: i) K is a subautomaton of G, and ii) if $s\in L(G)$ and $s\notin L(K)$, then there exists $\tilde{s} \in \overline{s}$ such that $\delta(\tilde{s},q_0)\notin X$.

The algorithm works under the assumption that $L_m(K)$ and $L_m(G)$ are closed.

Let us consider the automata pictured in Fig. 1, in which K is a subautomaton of G. However, K is not a strict-subautomaton of G because $ac\in L(G)$, $ac\notin L(K)$, $\overline{ac}=\{a,ac\}$, and neither $\delta(a,x_0)\notin X$ nor $\delta(ac,x_0)\notin X$. But K is a strict-subautomaton of G'. Given two automata K and G with $L(K)\subseteq L(G)$, it is always possible to construct automata K' and G' such that $L(K')=L(K)$, $L(G')=L(G)$, and K' is a strict-subautomaton of G' [7].

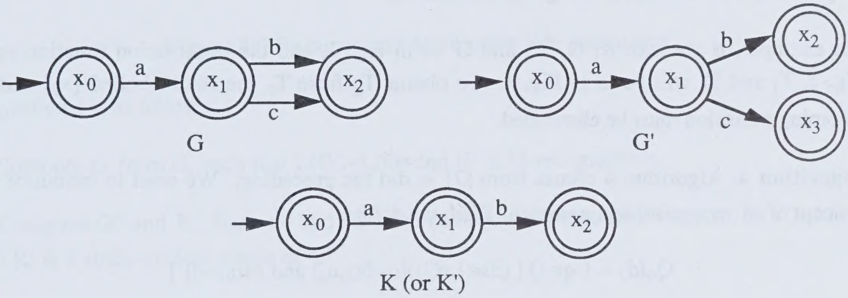
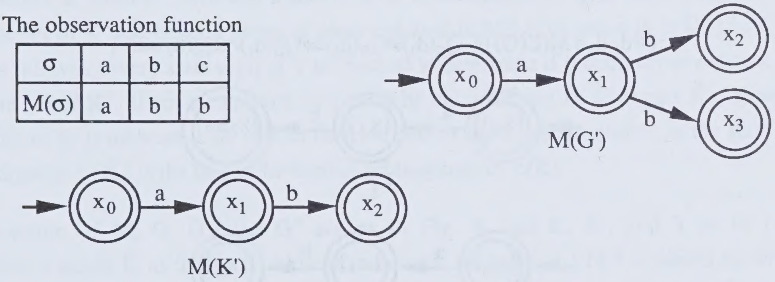


Figure 1: The notion of subautomaton


The observation function

σ	a	b	c
M(σ)	a	b	b

Figure 2: The observation function M and nondeterministic automata that generate M(L(G)) and M(L(K))

Algorithm 3 is based on the following observation. Let T_1 and T_2 be two deterministic automata that generate M(L(G)) and M(L(K)), respectively. If K is a strict-subautomaton of G, then L(K) is M-normal if and only if T_2 is a subautomaton of T_1.

The algorithm proceeds as follows:

1. Compute G' and K' such that L(G')=L(G), L(K')=L(K), and K' is a strict-subautomaton of G'.

2. Construct two deterministic automata T_1 and T_2 that generate M(L(G)) and M(L(K)), from G' and K'. The algorithm in [8] can be used for that purpose.

3. Compute an automaton $\tilde{T}_2$ resulting from T_2 but with states and transitions eliminated such that $\tilde{T}_2$ is the largest subautomaton of T_1 that is also a subautomaton of T_2. We have that $L(\tilde{T}_2)=M(L_R)$.

4. Compute an automaton T_3 such that $L(T_3)=L(K)\cap M^{-1}(L(\tilde{T}_2))$. An algorithm for that purpose is presented in [7]. Language $L(T_3)$ is L_R.

For example, if we take K, G, K', and G' as in Fig. 1, and the observation function as in Fig. 2, T_1 and T_2 will be as in Fig. 3. To obtain $\tilde{T}_2$ from T_2, the state labelled $\{x_2\}$ and its incoming transition must be eliminated.

Algorithm 4. Algorithm 4 comes from [7] as did the preceding. We need to introduce the concept of M-recognizable automaton. First, we define:

$$Q_G(d) = \{\, q\in Q \mid (\exists s\in L(G))[q=\delta(s,q_0) \text{ and } M(s)=d] \,\}$$

Set $Q_G(d)$ is defined as the set of states of G reachable with strings observed as d. Automaton G is *M-recognizable* if:

$$(\forall d_1,d_2\in M(L(G)))[\, Q_G(d_1)\neq Q_G(d_2) \Rightarrow Q_G(d_1)\cap Q_G(d_2)=\varnothing \,]$$

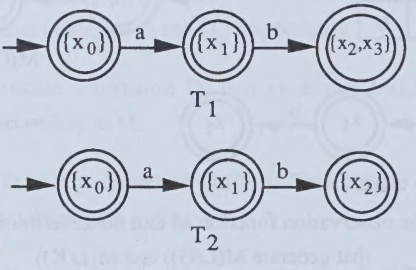


Figure 3: Deterministic automata T_1 and T_2 that generate M(L(G)) and M(L(K))

Note that it is always possible to construct an M-recognizable automaton G' from G such that L(G')=L(G) (see [7] for the procedure). For example, in Fig. 4, automaton G is not M-recognizable, because $Q_G(a)=\{x_1\}$ and $Q_G(b)=\{x_1,x_2\}$, hence $Q_G(a)\neq Q_G(b)$ and $Q_G(a)\cap Q_G(b)\neq\varnothing$. However, automaton G' is M-recognizable.

The algorithm works under the assumption that $L_m(K)$ and $L_m(G)$ are closed.

The observation function

σ	a	b	c
$M(\sigma)$	a	b	b

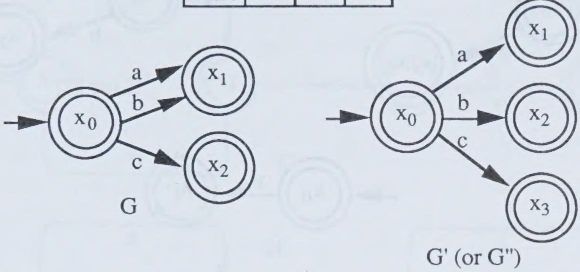


Figure 4: The concept of M-recognizable automaton

Algorithm 4 is as follows:

1. Compute G' from G, such that L(G')=L(G) and G' is M-recognizable.

2. Compute G" and K', from G' and K, respectively, such that L(G")=L(G'), L(K')=L(K), and K' is a strict-subautomaton of G".

3. Construct T, a deterministic automaton that generates M(L(G")), from G".

4. Construct $\tilde{K}$ from K', such that a state x of K' is inserted in $\tilde{K}$ if all states in $Q_{G''}(d)$ are in the state set of K' whenever a string is observed as d in M(L(G)) and x is in $Q_{G''}(d)$. This is done as follows. Every state set q of T is checked to determine if it contains any state x not in the state set of K'. If so, every state in q must be deleted from K' to obtain $\tilde{K}$. Finally, a transition of K' is inserted in $\tilde{K}$ if both its from and to states are admissible in the state set of $\tilde{K}$. Language $L(\tilde{K})$ is the largest M-normal sublanguage of L(K).

For example, if M, G, G', and G" are as in Fig. 4, and K, K', and T as in Fig. 5, Algorithm 4 yields $\tilde{K}$ as in Fig. 5. Note that the state set $q=\{x_2,x_3\}$ of T contains x_3, which is not in the state set of K'. Consequently, x_2 must be deleted from K' to obtain $\tilde{K}$, and the transition on event b from state x_0 to state x_2 in K' must also be rejected.

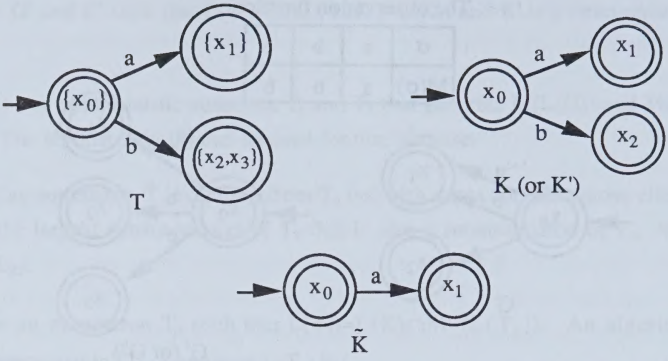


Figure 5: Automata K, K', T, and $\tilde{K}$

Algorithm 5. The fifth algorithm is taken from [9]. It works under the assumption that $L_m(K)$ and $L_m(G)$ are closed.

Algorithm 5 proceeds as follows:

1. Compute G' and K' such that L(G')=L(G), L(K')=L(K), and K' is a strict-subautomaton of G'.

2. From K', construct an automaton C_1, which is the translation of M(K') into a deterministic automaton. The states of C_1 are subsets of the states of K', and $L(C_1)=M(L)$. The algorithm of Lewis and Papadimitriou [8] can be used for this purpose.

3. Build a second automaton C_2, which is obtained from C_1 by pruning states and transitions. The pruned states are those for which so-called nonrecognizable (a concept defined below) ε-transitions are active, and the pruned transitions do not possess the recognizable property. The language of C_2 is $M(L_R)$.

4. Compute an automaton C_3 such that $L(C_3)=L(K)\cap M^{-1}(L(C_2))$ (see [7] for the algorithm). Language $L(C_3)$ is L_R.

A transition of C_1 from a state y to a state y' labelled with an event $\lambda \in (\Lambda \cup \{\in\})$ is *recognizable* if there is no state $x_i \in y$ and event $\tau \in \Sigma$ such that: i) in state x_i of the plant, event τ is active, ii) event τ is observed as λ, and iii) event τ is not admitted from x_i.

Referring to Figs. 6 and 7, it is obvious that *recognizable*(y_i,ε) returns true for i=0,1. However, this is not the case for i=2, since we have $\delta(f,x_3)$ defined, $M(f)=\varepsilon$, and $\xi(f,x_3)$ undefined. Hence, state y_2 must be eliminated as well as the incoming transition on event e. Similarly, *recognizable*(y_0,a) is true. However, *recognizable*(y_0,b) is false because we have

$\delta(c,x_0)$ defined, but $M(c)=M(b)$ and $\xi(c,x_0)$ undefined. Thus, the transition from y_0 to y_1 on event b must be removed. The resulting automaton C_2 is shown in Fig. 8.

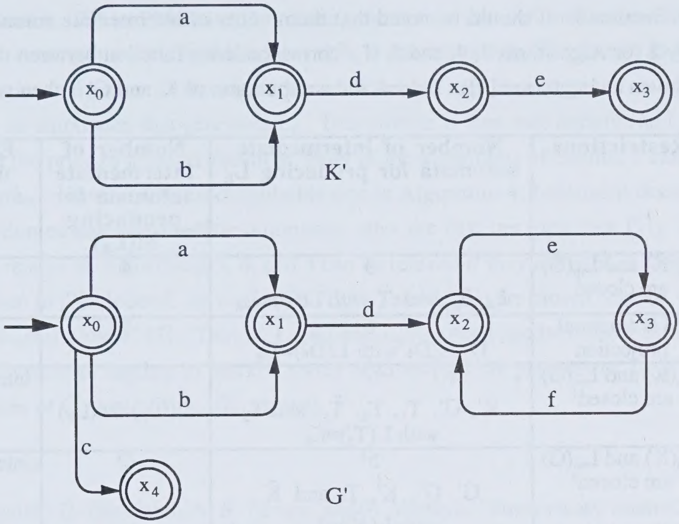

a
x0
x1
d
x2
e
x3
b
K'
a
e
x0
x1
d
x2
x3
b
f
c
x4
G'

Figure 6: Automata K' and G'

The observation function

σ	a	b	c	d	e	f
M(σ)	a	b	b	ε	e	ε

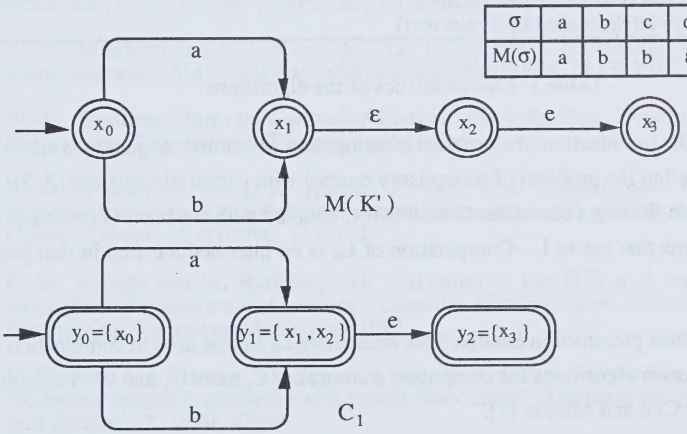

a
x0
x1
ε
x2
e
x3
b
M(K')
a
y0={x0}
y1={ x1 , x2 }
e
y2={x3 }
b
C1

Figure 7: Automata that generate M(K') and C_1

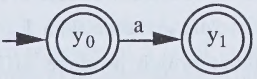


Figure 8: Automaton C_2

4. Comparison of the algorithms and conclusions

Table 1 gives the results of an analysis comparing characteristics of the five algorithms presented in Section 3. It should be noted that the number of intermediate automata can be decreased by 2 for Algorithms 3, 4, and 5, if a correspondence function between the states of K and G is given as input (see [9]). Indeed, the computation of K' and G' is then superfluous.

Alg.	Restrictions	Number of intermediate automata for producing L_R	Number of intermediate automata for producing $M(L_R)$	Ease of under-standing
1	$L_m(K)$ and $L_m(G)$ are closed	3 ξ_R, δ_R, and $\hat{T}$ with $L(\hat{T})=L_R$	4	Low
2	M is a natural projection	6 $D_1,\ldots,D_6$ with $L(D_6)=L_R$	7	High
3	$L_m(K)$ and $L_m(G)$ are closed[†]	6[‡] K', G', T_1, T_2, $\tilde{T}_2$, and T_3 with $L(T_3)=L_R$	5[‡] $L(\tilde{T}_2)=M(L_R)$	Intermediate
4	$L_m(K)$ and $L_m(G)$ are closed[†]	5[‡] G', G'', K', T, and $\tilde{K}$ with $L(\tilde{K})=L_R$	6[‡]	Intermediate
5	$L_m(K)$ and $L_m(G)$ are closed[†]	5[‡] K', G', C_1, C_2, and C_3 with $L(C_3)=L_R$	4[‡] $L(C_2)=M(L_R)$	Intermediate

[†] This constraint can be relaxed (see text).
[‡] This number can be decreased by 2 (see text).

Table 1: Characteristics of the algorithms

As mentioned in the introduction, the problem of computing M-normal languages is significant because it is related to the problem of supervisory control with partial observation [2, 3]. The problem consists in finding a controller C such that C coupled with the plant G generates only sequences of events that are in L. Computation of L_R is an intermediate step in that process (see [1,3-7,9]).

Any of the algorithms presented in Section 3 for computing L_R can be used in combination with either of the two main algorithms for computing a controller C, namely, that of Wonham and Ramadge [3], and Cho and Marcus [7].

The algorithm in [3] works under the assumption that $M^{-1}(M(\Sigma_u)-\{\varepsilon\})\subseteq\Sigma_u$. That is, no controllable event can be observed as an uncontrollable event [1]. Furthermore, it takes as input the projection of L_R, that is, $M(L_R)$ rather than L_R. Therefore, the most suitable algorithms in that context are those which produce $M(L_R)$ with less effort. They are Algorithms 3 and 5, without their respective last step, because they compute an automaton that generates $M(L_R)$ through the construction of only 5 and 4 automata respectively (3 and 2, if a correspondence function has already been constructed).

The algorithm in Cho and Marcus [7] is more general than that in Wonham and Ramadge [3]. Indeed, the assumption that $M^{-1}(M(\Sigma_u)-\{\varepsilon\})\subseteq\Sigma_u$ is not necessary. The former is based on the application in alternation of the algorithm of Wonham and Ramadge and an algorithm for computing an automaton that generates L_R. This process is therefore iterative and generality is obtained at the price of a longer run-time. Any of the algorithms of Section 3 can be used for computing L_R. However, the most suitable one is Algorithm 4, because it does not require reconstruction of any intermediate automaton after the first iteration (see [7]). Finally, the restriction related to Algorithms 3, 4, and 5 can be relaxed if they are used in alternation with the algorithm in [7]. Indeed, the constraint "$L_m(K)$ and $L_m(G)$ are closed" can be weakened to "$L_m(K)$ is $L_m(G)$-closed" [7]. That is, $L_m(K)$ and $L_m(G)$ need not be closed, but $L_m(K)$ must contain only strings leading to marked states of G and all the prefixes of $L_m(K)$ leading to marked states of G must be in $L_m(K)$ as well.

References

[1] R. Cieslak, C. Desclaux, A. S. Fawaz, and P. Varaiya, "Supervisory control of discrete-event processes with partial observations", *IEEE Transactions on Automatic Control*, 33 (3) 1988, 249-260.

[2] P. J. Ramadge and W. M. Wonham, "Supervisory control of a class of discrete event processes", *SIAM Journal Control and Optimization*, 25 (1) 1987, 206-230.

[3] W. M. Wonham and P. J. Ramadge, "On the supremal controllable sublanguage of a given language", *SIAM Journal Control and Optimization*, 25 (3) 1987, 637-659.

[4] W. M. Wonham, *Notes on Control of Discrete Event Systems*, Department of Electrical Engineering, University of Toronto, June 1993.

[5] R. D. Brandt, V. Garg, R. Kumar, F. Lin, S. I. Marcus, and W. M. Wonham, "Formulas for calculating supremal controllable and normal sublanguages", *Systems and Control Letters*, 15 (2) 1990, 111-117.

[6] C. M. Enright and M. Barbeau, "An evaluation of the TCT tool for the synthesis of controllers of discrete event systems", Canadian Conference on Electrical and Computer Engineering, Vancouver, September 1993.

[7] H. Cho and S. I. Marcus, "On supremal languages of classes of sublanguages that arise in supervisor synthesis problems with partial observation", *Mathematics of Control, Signals, and Systems*, 2, 1989, 47-69.

[8] H. R. Lewis and C. H. Papadimitriou, *Elements of the Theory of Computation*, Prentice-Hall, 1981.

[9] M. Barbeau, G. Custeau, and R. St-Denis, "An algorithm for computing the projection of the largest recognizable sublanguage of a legal language", Technical report no. 111, Département de mathématiques et d'informatique, Université de Sherbrooke, April 1993 (submitted for publication).

Controlled Stochastic Languages

H. Mortazavian

Department of Electrical and Computer Engineering
Wayne State Universtiy
Detroit, MI 48202
email: hmor@eng.wayne.edu

Keywords: Discrete event systems, supervisory control, stochastic automata, stochastic languages, optimal control.

1. Introduction

In [9] we presented a preliminary version of an approach to stochastic supervisory control of discrete event systems. This paper continues along the lines of the previous paper. Currently there is significant interest in the study of stochastic discrete event systems [5,6,9] . In this paper we present necesessary and sufficient conditions for existence of a supervisor that operates in a stochastic environment. Several other modeling and control problems are considered.

2. Stochastic Discrete Event System

We define a stochastic discrete event system G_s as a triple

$$G_s = (\Sigma, L, \mu)$$

where $L \subseteq \Sigma^*$ is a language and $\mu : L \times \Sigma \rightarrow [0, 1]$ is a discrete probability measure, so that for $(s, \sigma) \; \varepsilon \; L \times \Sigma$, we have

(i) $\mu(s, \sigma) \geq 0$; (ii) $\Sigma_{\sigma \varepsilon \Sigma} \; \mu(s,\sigma) = 1$

We assume, in general, that L is a closed language. Thus a stochastic DES is a deterministic DES with a stochastic structure imposed on it through the measure μ. With the process having behaved as s, the value of $\mu(s, \sigma)$ is the (conditional) probability of occurrence of the event σ. Thus μ distributes the probability mass over the alphabet, as a function of the history, as the process evolves. In order to calculate the value of μ for each σ, we need to know s partly or completely. In some *regenerative processes,* we need only consider $\mu(\sigma_n \mid \sigma_{n-1}, ..., \sigma_{n-k})$ which in the Markovian case reduces to $\mu(\sigma_n \mid \sigma_{n-1})$. Clearly $P(\sigma | s) = P(\sigma | s\sigma \varepsilon L)$. Initially, when no events have yet occurred the set $\{ \mu(\sigma | \sigma \varepsilon L)\}$ defines an initial distribution on Σ. If in (ii) the equality is replaced by an inequality, a *subprobability measure* is defined, with the interpretation that the process having performed s may terminate.

With process G_s we may associate a discrete measure $\varphi : L \rightarrow [0, 1]$ so that (i) $\varphi(s) = 0$ if $s \notin L$; (ii) $0 \leq \varphi(s) \leq 1$ if $s \in L$; (iii) $\Sigma_{s \varepsilon L} \varphi(s) = 1$; $\forall s \varepsilon \Sigma^*$

Clearly, the value of $\varphi(s)$ is the probability that $s \varepsilon L$. Again, if we relax the equality in (iii) we have a subprobability measure, in which case the process is allowed to terminate after each s. We remark in passing that when L is an infinite language one can show that

$$(\forall \varepsilon > 0)\ (\exists N \geq 0)\ l(s) \geq N \Rightarrow \varphi(s) < \varepsilon$$

i.e., if there are any "very long" strings in the language their probability is "very small". Clearly , (i) $\varphi(\varepsilon) = 1$; (ii) $\varphi(s) = \Sigma_{\sigma \varepsilon \Sigma} \varphi(s\sigma)$, $\forall s \varepsilon L$

In the case of terminating processes the equality in (ii) is replaced by an inequality . Note also that for $s \leq t$ we have $\varphi(t) \leq \varphi(s)$.

Example [19]. Let $L = \{ \alpha^n \beta^n ;\ n \geq 0\}$ be a language over $\{\alpha , \beta\}$, and define $\varphi(\alpha^n \beta^n) = 1 / e.n!$; $\varphi(s) = 0$ for $s \notin L$. Clearly, conditions (i), (ii), (iii) in the definition of φ are satisfied, so that this language is stochastic.

The triple (Σ, L, φ) may be alternatively taken as the definition of G_s . We remark in passing that if instead of a language L one uses its characteristic function, then the triples defined so far may be represented as (Σ, P, μ) and (Σ, P, φ) where $P : \Sigma^* \rightarrow [0, 1]$ is a predicate on Σ^*. The interpretation being that a string $s \varepsilon \Sigma^*$ is generated by G_s iff $P(s) : L(G) = \{ s \varepsilon \Sigma^* : P(s) \}$; i.e., the behavior of G is described by strings s all whose prefixes satisfy P, so that L(G) is a closed language. The predicate P abstractly represents the dynamics of the system (for further work along this line see [8]).The language L or the predicate P can be represented by automata, Petri nets, fixed point equations, etc. A final remark regarding the definition of a stochastic DES is in order. We may, in defining $G_s = (\Sigma, L, \varphi)$ refrain from explicit reference to L altogether and simply define G_s as a pair (Σ, φ) where $\varphi : \Sigma^* \rightarrow [0, 1]$ is a discrete measure. Then the passage from a deterministic process $G = (\Sigma, P)$ to a stochastic process $Gs = (\Sigma, P)$ amounts to extending the range of the characterizing mapping of the system from the pair {0, 1} to the interval [0, 1], i.e., replacing a deterministic function by a stochastic function. A similar mapping φ was introduced by Zadeh to define Fuzzy sets. But whereas the value of $\varphi(s)$ is here interpreted as the probability of the process behaving as s, in Fuzzy set theory this is thought of as "degree of membership" of string in a language. We shall not pursue this discussion further here .

Let φ and ψ be stochastic mappings : $\varphi, \psi : \Sigma^* \rightarrow [0, 1]$. We define

$$(\alpha \varphi + \beta \psi)(s) = \alpha \varphi(s) + \beta \psi(s), \quad \alpha, \beta \geq 0; \ \alpha + \beta \leq 1$$

$$(\varphi . \psi)(s) = \varphi(s) \psi(s)$$

$$(\varphi \vee \psi)(s) = \max(\psi(s), \psi(s))$$

$$(\varphi \wedge \psi)(s) = \min(\varphi(s), \psi(s))$$

$$(\varphi \Delta \psi)(s) = \varphi \times \psi(s) = \varphi(s) + \psi(s) - \varphi(s) \psi(s)$$

$$(\psi \oplus \psi)(s) = \min(1, \varphi(s) + \psi(s))$$

$$(\varphi \otimes \psi)(s) \backslash = \min(1, \Sigma_{rs=P} \varphi(r) \psi(s))$$

$$(\varphi)^*(s) = \begin{array}{ll} 1, & \text{if } s = \varepsilon \\ \min(1, \sum\limits_{m=1} \sum\limits_{\substack{s1,\dots,sm \neq \varepsilon \\ s1,\dots,sm = s}} \prod \varphi(s_i)), & \text{if } s \neq \varepsilon \end{array}$$

These operations show how two stochastic DES or their languages may be combined[13].

3. Preliminaries on Supervisory Control

Let Σ be a finite nonempty set, called *alphabet*. Let Σ^* denote the set of all finite strings $s = \sigma_1 \sigma_2 \ldots \sigma_n$ of elements $\sigma_i \varepsilon \Sigma$ and include in Σ^* the empty string ε. A string t is called a *prefix* of a string s (denoted by $t \leq s$) if for some string r the equation $tr = s$ holds. When $r \neq \varepsilon$, t is called a *proper prefix* of s (denoted by $r < t$). A *language* over Σ is any subset of Σ^*. The *prefix closure* L' of a language L is defined by $L = \{ s : st \varepsilon L \text{ for some } t \varepsilon \Sigma^* \}$ i.e.,L' is the set of all prefixes of the strings in L. L is prefix closed if $L = L'$.

A *nondeterministic automaton* is a sextuple $A = (\Sigma , Q , \delta , \Gamma, \gamma , Q_o)$ where Q is a nonempty set of states, Σ is a nonempty input alphabet, Γ is a nonempty output alphabet, $\delta : \Sigma \times Q \rightarrow 2^Q$ is the transition function, $\gamma : Q \rightarrow \Gamma$ is the output function, and $Q_o \subseteq Q$ is the set of initial states. An automaton is called *sequential deterministic* if $\delta(\sigma, q)$ always has *exactly* one element. If we leave out all reference to input (or output) alphabet we define an automaton with no input (output) .

In supervisory control [1,7,12,20] a DES is modeled by a finite state deterministic automaton $G = (\Sigma , Q , \delta , q_o , Q_m)$ where Σ is the alphabet, Q is the finite set of states, $\delta : \Sigma \times Q \rightarrow Q$ is the transition function, $q_o \varepsilon Q$ is the initial state and Q_m a set of "marked" or final states. The set of events is partitioned as $\Sigma = \Sigma_c \cup \Sigma_{uc} = \Sigma_o \cup \Sigma_{uo}$, where Σ_c, Σ_{uc}, Σ_o and Σ_{uo} denote the subset of controllable , uncontrollable, observable and unobservable events, respectively.

A supervisor is a pair $S = (S , \psi)$ where $S = (\Sigma , X , \xi , x_o , X_m)$ is a finite state deterministic automaton with transition function $\xi : \Sigma \times X \rightarrow X$ satisfying $\xi (\sigma , x) = x$ for all $\sigma \varepsilon \Sigma_{uc}$ and $\psi : \Sigma \times X \rightarrow \{ 0, 1\}$ is a map satisfying

$\psi (\sigma , x) = 1$ if $\sigma \varepsilon \Sigma_{uc}$, $x \varepsilon X$; $\psi (\sigma , x) \varepsilon \{ 0, 1\}$ if $\sigma \varepsilon \Sigma_c$, $x \varepsilon X$

$\psi (\sigma, x) =1$ means that σ is enabled a ; $\psi(\sigma ,x)= 0$ means that σ is disabled at x.

Coupling the supervisor with the process results in a controlled structure

$$(S, G) = (\Sigma, X \times Q, (\xi, \delta), (x_o, q_o), X_m \times Q_m)$$

where $(\xi \times \delta) : \Sigma \times X \times Q \rightarrow X \times Q$ is defined as

$$(\xi\times\delta)(\sigma,x,q)=\begin{cases}(\xi(\sigma, x), \delta(\sigma, q)) & \text{if } \psi(\sigma, x) =1\\ \text{undefined} & \text{if } \psi(\sigma, x) =0\end{cases}$$

(S, G) is itself an automaton whose behavior is described by $L(S, G)$. The language marked by the controlled system is denoted $L_m(S, G)$. Necessary and sufficient conditions for existence of supervisors are given in terms of the notions of controllability and observability of a language.

4. Stochastic Automata

A *Finite stochastic automaton* over the alphabet Σ is a quadruple

$$G_s = (Q, M, \pi_o, F)$$

where $Q = \{q_1,..., q_n\}$ is a finite nonempty set of states, M is a mapping of Σ into the set of stochastic $n \times n$ matrices, π_0 is an n-dimensional stochastic row vector (the initial distribution) and F is a subset of Q called the set of final states. To any string $s \varepsilon \Sigma^*$ we associate an $n \times n$ stochastic matrix by defining

(i) $M(\varepsilon) = I$ (the $n \times n$ identity matrix)

(ii) $M(s\sigma) = M(s)\, M(\sigma), \quad s \varepsilon \Sigma^*, \sigma \varepsilon \Sigma$

Thus we extend the domain of the mapping M from Σ to Σ^*. Clearly

(ii)' $M(\sigma_1 \sigma_2 .... \sigma_n) = M(\sigma_1)\, M(\sigma_2) ... M(\sigma_n)$

Let π_f be the n-dimensional column vector whose ith component equals 1 if $q_i \varepsilon F$ and 0 otherwise. For a real number λ, $0 \leq \lambda < 1$, the language *accepted* by G_s with the *cut-point* λ is defined by

$$L(G_s, \lambda) = \{ s \varepsilon \Sigma^* : \pi_o M(s) \pi_f > \lambda\}$$

Sometimes we write

$$G_s = (\pi_o, M(\sigma), \pi_f)$$

and define a stochastic language as a map $\varphi : \Sigma^* \rightarrow [0, 1]$ if there exists a stochastic automaton G_s, for which $\varphi(s) = \pi_o M(s) \pi_f$. Alternatively, we may take the characteristic set of this map, namely $L(G_s) \subseteq \Sigma^*$ and call it stochastic if there exists a map φ as defined, and a real number λ, $0 \leq \lambda < 1$, such that

$$L(G_s) = L(G_s, \lambda) = \{ s : \varphi(s) > \lambda \}.$$

Rabin [18] first introduced automata without reference to an initial distribution, thus specifying as initial state q_o. In this case one may define M as the map $M : Q \times \Sigma \rightarrow \{(x_o, ..., x_n)\} ;\ x_i \, \varepsilon \, [0, 1]$

such that for

$(q, \sigma) \, \varepsilon \, Q \times \Sigma$,

$M(q, \sigma) = (P_o(q, \sigma), ..., P_n(q, \sigma)), P_i(q, \sigma) \geq 0 ;\ \Sigma_i P_i(q, \sigma) = 1.$

When the automaton is in state q with the event σ it can go into any one of the states $q_i \, \varepsilon \, Q$ with probability $P_i(q, \sigma)$. For $\sigma \, \varepsilon \, \Sigma$ and $s = \sigma_1 ... \sigma_m$ define the $n \times 1$ by $n \times 1$ matrices $M(\sigma)$ and $M(s)$ by

$$M(\sigma) = [P_j(q_i, \sigma)] \quad 0 \leq i \leq n, 0 \leq j \leq n$$

$$M(s) = M(\sigma_1) M(\sigma_2) ... M(\sigma_m) = [P_j(q_i, \sigma)]$$

The probability that G_s will move from state q_i to state q_j via the string s is the $(i+1, j+1)$ element $p_j(q_i, s)$. Let $F = \{ q_{io}, ..., q_{ir} \}$ and $I = \{ io, ..., ir \}$. Define $P(s) = \Sigma_{i \varepsilon I} P_i(q_o, s)$; i.e., the probability for G_s, when started in q_o to enter into a state in F following the sequence of transitions s. The language $L(G_s, \lambda)$ is then defined as $L(G_s, \lambda) = \{ s \, \varepsilon \, \Sigma^* : P(s) > \lambda \}$. The string s is said to be *accepted* by G_s with cut-point λ, if $s \, \varepsilon \, L(G_s, \lambda)$ and $L(G_s, \lambda)$ is called the language defined by G_s with cut-point λ.

If $M(q, \sigma) = q_i$ then G_s can be viewed as if it will enter state q_i with probability 1. It is well known that stochastic automata define a strictly larger class of languages [2,10,11,13,114,15,16,17,].

5. Stochastic Controllability and Observability of Languages

In this Section we introduce the concepts of stochastic controllability and stochastic observability of languages.

We say a language $K \subseteq L \subseteq \Sigma^*$ is *stochastically controllable* with respect to a language L and uncontrollable events $\Sigma_{uc} \subseteq \Sigma$, with a parameter λ, $0 \leq \lambda < 1$ if for all $s \varepsilon K$ and for all $\sigma \varepsilon \Sigma_{uc}$

$$P[s\sigma \varepsilon K \mid s \varepsilon K \wedge \sigma \varepsilon \Sigma_{uc} \wedge s\sigma \varepsilon L(G)] > \lambda$$

If we call a string "legal" if it belongs to K, a language is stochastically controllable with parameter λ, if the probability that uncontrollable transitions leave legal strings legal exceeds λ. Thus, the definition implies a *stochastic invariance* property under uncontrollable transitions.

Let Σ_o, $\Sigma_{uo} \subseteq \Sigma$ denote the subsets of observable and unobservable events, respectively. Define the projection $\mathbb{P} : \Sigma^* \rightarrow \Sigma_o{}^*$ by (i) $\mathbb{P}\varepsilon = \varepsilon$; (ii) $\mathbb{P}\sigma = \varepsilon$ if $\sigma \varepsilon \Sigma_{uo}$, $\mathbb{P}\sigma = \sigma$ if $\sigma \varepsilon \Sigma_o$; (iii) $\mathbb{P}(s\sigma) = \mathbb{P}(s)\ \mathbb{P}(\sigma)$ for $s \varepsilon \Sigma^*$, $\sigma \varepsilon \Sigma$. Thus the effect of $\mathbb{P}$ on a string is just to erase its unobservable symbols.

We say a language $K \subseteq L \subseteq \Sigma^*$ is *stochastically observable* with respect to a language L and unobservable events $\Sigma_{uo} \subseteq \Sigma$, with a parameter λ, $0 \leq \lambda < 1$, if

$$(\forall s, s' \varepsilon \Sigma^*)\ Ps = Ps' \Rightarrow (\forall \sigma)\ P[s'\sigma \varepsilon K \mid s\sigma \varepsilon K \wedge s' \varepsilon K \wedge s'\sigma \varepsilon L(G)] > \lambda$$

For a closed language $L \subseteq L(G)$, we define the set of closed and λ – *controllable sublanguages* of L as $\underline{C}(L, \lambda) = \{K : K \subseteq L \;\&\; K$ is closed and λ - controllable$\}$.

Proposition 1. The set $\underline{C}(L, \lambda)$ is closed under union. That is, if $K_1, K_2 \ \varepsilon\ C(L, \lambda)$ then $K_1 \cup K_2 \ \varepsilon\ C(L, \lambda)$.

Since $\underline{C}(L, \lambda)$ is closed under union, there exists a supremal element sup $\underline{C}(L, \lambda)$, called the *supremal* λ - *controllable sublanguage* of L. That is, sup $\underline{C}(L, \lambda) \ \varepsilon\ \underline{C}(L, \lambda)$ for all $K \ \varepsilon\ \underline{C}(L, \lambda)$, $K \subset \sup \underline{C}(L, \lambda)$.

For a closed language $L \subseteq L(G)$ we define the set of *closed* and λ - *observable superlanguages* of L as $\underline{O}(L, \lambda) = \{K : K \supseteq L \;\&\; L$ is closed & λ – observable$\}$.

Propositon 2. The set $\underline{O}(L, \lambda)$ is closed under intersection. That is, if $K_1, K_2 \varepsilon \underline{O}(L, \lambda)$ then $K_1, K_2 \varepsilon \underline{O}(L, \lambda)$.

Since $\underline{O}(L, \lambda)$ is closed under intersection, there is an infimal element inf $\underline{O}(L, \lambda)$, which we call the *infimal λ-observable superlanguage* of L.That is, inf $\underline{O}(L, \lambda) \varepsilon \underline{O}(L, \lambda)$ for all $K \varepsilon \underline{O}(L, \lambda), K \supseteq \inf \underline{O}(L, \lambda)$.

6. Necessary and Sufficient Conditions for Existence of a Supervisor in A Stochastic Environment

Let G_s be a deterministic generator with a probability structure imposed on it, and let S be a deterministic supervisor. We then have:

Theorem 1. Let K be a nonempty closed sublanguage of L (G). If there exists a supervisor S such that for $0 \leq \lambda < 1$

(i) $K \subseteq L(S, G)$

(ii) $P[L(S, G) \subseteq K] > \lambda$

then, K is λ-controllable.

Theorem 2. [9] Let K be a nonempty closed sublanguage of L (G). If there exists a (complete) supervisor S such that for $0 \leq \lambda < 1$

$P[L(S, G) = K] > \lambda$

then, K is λ-observable.

Proof. If K is not λ-observable, then

$(\exists s, s' \varepsilon \Sigma^*)(Ps = Ps') \wedge (\exists \sigma \varepsilon \Sigma)$

$P[s'\sigma \varepsilon K \mid s\sigma \varepsilon K \wedge s' \varepsilon K \wedge s'\sigma \varepsilon L(G)] \leq \lambda$ Now $Ps = Ps'$

$\Rightarrow \xi(s, x_o) = \xi(s', x_o)$

$\Rightarrow \psi(\sigma, \xi(s, x_o)) = \psi(\sigma, \xi(s', x_o))$

Case 1. $\psi(\sigma, \xi(s, x_o)) = \psi(\sigma, \xi(s', x_o)) = 1$

$\Rightarrow s'\sigma \varepsilon L(s, G) \wedge P[s'\sigma \varepsilon K] \leq \lambda$

Case 2 $\psi(\sigma, \xi(s, x_o)) = \psi(\sigma, \xi(s', x_o)) = 0$

$\Rightarrow s\sigma \notin L(s, G) \wedge P[s\sigma \varepsilon K] > \lambda$

In either case, $P[L(S,G) \neq K] > \lambda$ which contradicts the assumption of the theorem. Hence, K is λ - observable .

Theorem 3. Suppose K is λ-controllable and λ-observable. Then there exists a supervisor S such that

$$P[L(S,G) = K] > \lambda.$$

7.References

[1] Cieslak R. , C. Desclaux , A. Fawaz and P. Varaiya [1988], "Supervisory control of discrete-event processes with partial observations", *IEEE Trans. on Automatic Control* , 33(3) pp. 249-260.

[2] Doberkat E. [1981], *Stochastic Automata : Stability, Nondeterminism and Prediction,* Lecture Notes in Computer Science 113, Springer- Verlag, 1981.

[3] Fliess M. [1973], "Proprietes booleennes des langages stochastiques, " *Mathematical Systems Theory,* 7(4), pp. 353-359.

[4] Fliess M. [1975] , "Series rationnelles positives et processus stochastiques" , *Ann. Inst. Henri Poincare.* Sect B 11(2), 1-21.

[5] Garg V.K. [1992] , "An algebraic approach to modeling of probabilistic discrete event systems" *Proc. of the 31st IEEE Conf. on Decision and Control,* December, 2348-2353.

[6] Lawford M. and M.W. Wonham [1993], "Supervisory control of probabilistic discrete event systems", *Proc. of the 36th Midwest Symposium on Circuits and Systems* , August, To appear.

[7] Lin F. and W.M. Wonham [1988] "On observability of discrete event systems", *Information Sciences* , 44(3) , 173-198.

[8] Mortazavian H. and F. Lin [1991], "Foundations of a logical theory of modeling and control of discrete event systems", Invited Paper. *Proc. of the IFAC International Symposium on Distributed Intelligence Systems*, Arlington, VA, August 13-15, pp. 19-24.

[9] Mortazavian H. [1993], "On control of stochastic discrete event systems", *Proc. of the 2nd Eureopean Control Conference* , Groningen, June 28- July 1, pp. 2191-2196. Invited.

[10] Paz A. [1971], *Introduction to Probabilistic Automata*, Academic Press.

[11] Rabin M.O. [1963], "Probabilistic automata", *Information and Control*", 6, pp. 230-245.

[12] Ramadge P.J. and W.M. Wonham [1987], "Supervisory control of a class of discrete event processes", *SIAM J. Control and Optimization* , 25(1), pp. 206-230.

[13] Starke P.H. [1972], *Abstract Automata* , North-Holland Publishing Co., Amsterdam.

[14] Turakainen P. [1968], "On stochastic languages", *Information and Control*, 12, pp. 304-313.

[15] Turakainen P. [1968], "On probabilistic automata and their generalizations", *Annals Acad. Sci. Fennicae*, Ser. A.I., no. 429

[16] Turakainen P. [1969], "Generalized automata and stochastic languages", *Proceedings of the American Mathematical Society*, 21, pp. 303-309.

[17] Turakainen P. [1971], "Some closure properties of the family of stochastic languages", *Information and Control,* 18, pp. 253-256.

[18] von Neumann J. [1956], "Probabilistic logic and synthesis of reliable organisms from unreliable components", *Annals of Mathematics Studies*, vol. 34, C.E. Shannon and J. McCarthy, eds., pp. 43-98, Princeton University Press, Princeton, New Jersey.

[19] Whetherell C.S. [1980], "Probabilistic languages, a review and some open questions", *ACM Computing Surveys* 12 , 361-379.

[20] Wonham W.M. and P.J. Ramadge [1987] , "On the supremal controllable sublanguage of a given language", *SIAM J. Control and Optimization,* 25(3), pp. 635-659.

On Acceleration of Stochastic Approximation for Optimizations of Discrete Event Systems [1]

I-Jeng Wang and Edwin K. P. Chong
School of Electrical Engineering
Purdue University
West Lafayette, IN 47907

Abstract

This paper studies the possibility of improving the performance of SA algorithms by the finite-horizon designs of step-size. A Robbins-Monro type of algorithm is considered. As a starting point, we simplify the problem by assuming that the derivative of the performance measure is linear and its estimates are unbiased and independent. Firstly, an optimal step-size is derived to minimize the finite-horizon mean square error assuming that the derivative function and the variance of the estimation noises are known. A heuristic step-size based on the optimal step-size is also proposed along with simulation results which compare its performance with that of a popular decreasing step-size. Finally, the proposed heuristic algorithm, combined with IPA techniques, is applied to the optimization of a simple M/M/1 queue to illustrate its performance.

1 Introduction

Consider a discrete event system (DES) with a performance measure $J(\theta)$ which depends on a scalar parameter θ. If an estimate of $\frac{d}{d\theta}J(\theta)$ is available, then *stochastic approximation* (SA) algorithms can be applied to find the zero of the derivative function and accomplish the local optimization of $J(\theta)$. With the recent development of efficient gradient estimation techniques such as *infinitesimal perturbation analysis* (IPA) and *likelihood ratio* (LR), SA has emerged as an possible approach for the optimizations of complicated DESs. Nevertheless, it is well known that these algorithms usually suffer from slow convergence in practice. Although the asymptotic behavior of SA algorithms have been extensively studied since the first SA algorithm was proposed [1], the finite-time performance of these algorithms are only understood through simulations of individual cases. Since the time horizons are inherently finite in real-world applications, the study of finite-time behavior is important from both practical and theoretical points of view. More importantly, algorithms designed under the finite-horizon scenario have the potential of performing "better" in practice.

This paper marks our first effort toward building up a theory for finite-horizon designs of SA algorithms. The objective of this paper is to illustrate the possibility of improving the performance of SA algorithms by finite-horizon designs of step-size. We study a simple Robbins–Monro type of SA algorithm

$$\theta_{n+1} = \pi_{n+1}[\theta_n - a_n \hat{h}_{n+1}], \tag{1}$$

where $\pi_{n+1}[\cdot]$ is a projection operator defined by

$$\pi_{n+1}[x] = \begin{cases} \theta_n, & \text{if } x \notin D, \\ x, & \text{otherwise.} \end{cases}$$

---

[1] This research is supported by the Purdue Research Foundation grant.

and $\hat{h}_{n+1}$ is an estimate of the derivative of $J(\cdot)$ at θ_n. D is a constraint set which the parameter θ is restricted to. Denoting $\frac{dJ}{d\theta}(\theta)$ as $h(\theta)$, the above recursion can be rewritten as

$$\theta_{n+1} = \pi_{n+1}[\theta_n - a_n(h(\theta_n) + \varepsilon_{n+1})],$$

where ε_{n+1} corresponds to the error of the nth derivative estimate. The asymptotic properties of this algorithm have been extensively studied [1, 2, 3].

In Section 2, we derive step-sizes for linear derivative functions according to a finite horizon criterion. Their performance is then compared with the performance of an asymptotically optimal step-size. We apply a heuristic step-size, combined with the IPA technique, to the optimization of the steady state mean system time of a M/M/1 queue in Section 3. Comparisons with several other algorithms are made through simulations to illustrate the performance of the proposed heuristic step-size. Finally, conclusions are drawn in Section 4.

2 Choices of Step-size: Linear Scalar Case

As a starting point, we consider the simplest case where the scalar derivative function $h(\cdot)$ is linear and the estimates $\{\hat{h}_{n+1}\}$ are independent and unbiased. The parameter θ is assumed to be unconstrained. More precisely, we deal with the scalar functions $h(\cdot)$ of the following form :

$$\begin{aligned} h(\theta) &= m\theta + b, \ m > 0, \\ h(\theta^*) &= 0, \end{aligned}$$

where θ^* is the zero of the function. Assume $\{\varepsilon_{n+1}\}$ are independent, zero mean with

$$E(\varepsilon_{n+1}^2) = v_{n+1} < \infty.$$

We also denote the σ-algebra generated by $\{\varepsilon_k : k \leq n\}$ as $\mathcal{F}_n$. Define the mean square error of the SA algorithm at step n as

$$V_n = E[(\theta_n - \theta^*)^2].$$

θ_1 is an arbitrary deterministic scalar which is given as the initial estimate of θ^*. The problem here is to find a sequence of step-sizes $\{a_n : 1 \leq n \leq N\}$ such that V_N resulting from the unconstrained SA algorithm (1) is minimized, where N is a prespecified positive integer which corresponds to the time horizon. Only the first and second order statistical characteristics of the noise $\{\varepsilon_{n+1}\}$ are available. The step-size a_n need not be fixed—it can be a random variable which is measurable with respect to $\mathcal{F}_{n+1}$. In other words, a_n might be an adaptive step-size which depends upon the available observations or estimates of the derivative function, $\{\hat{h}_{k+1} : k \leq n\}$, and the past history of the SA algorithm, $\{\theta_k : k \leq n\}$. At the first glance, the problem described above is very similar to a finite-horizon *Dynamic Programming* (DP) problem. However, they are quite different in two aspects. First, the step-size a_n, which corresponds to the control applied at n in a DP setting, is measurable with respect to ε_{n+1}. On the contrary, in a DP setting ε_{n+1} is considered available only after the control a_n is applied. Besides that, the dynamics of the state equation which totally depend on the derivative function $h(\cdot)$ in this case, and the statistical characteristics of the noise are usually known for a standard DP problem. This is not true for the problems that SA algorithms are meant to deal with.

2.1 Deterministic Step-sizes

A typical deterministic step-size is $a_n = an^{-1}$ for some constant $a > 0$. It has been shown by several authors [4, 5, 6] that, by taking $a = \frac{1}{h'(\theta^*)}$, the variance of the limiting normal distribution of the error is minimized. Therefore, we can consider this step-size $a_n^* = \frac{1}{h'(\theta^*)}n^{-1}$ as an *asymptotically optimal* step-size. We use a_n^* as a reference for comparison with the finite horizon step-size which we shall derive in the following.

Assume the step-size is deterministic. Then the recursion for the mean square error, V_n, can be obtained as

$$V_{n+1} = (1 - a_n m)^2 V_n + a_n^2 v_{n+1}. \tag{2}$$

Since $V_n = E[h(\theta_n)^2]/m^2$, it is equivalent to work with $E[h(\theta_n)^2]$. Defining $x_n = E[h(\theta_n)^2]$, the following can be derived from (2)

$$x_{n+1} = (1 - a_n m)^2 x_n + a_n^2 m^2 v_{n+1}. \tag{3}$$

The problem then can be solved by applying the *Dynamic Programming* (DP) algorithm [7]. The optimal deterministic step-size obtained is

$$a_n = \frac{mV_n}{V_n m^2 + v_{n+1}}, \quad 1 \leq n < N. \tag{4}$$

The optimal cost associated with (4) is

$$V_N = \frac{v_2}{(1 + \sum_{k=2}^{N-1} \frac{v_{k+1}}{v_k})m^2 + \frac{v_3}{V_1}},$$

where $V_1 = (\theta_1 - \theta^*)^2$.

A recursive equation for this step-size can be derived from (3) and (4) :

$$\begin{aligned} a_1 &= \frac{h(\theta_1)}{m[h(\theta_1)^2 + v_2]}, \\ a_{n+1} &= \frac{v_{n+1} a_n}{m v_{n+1} a_n + v_{n+2}}. \end{aligned}$$

Further suppose the noise is i.i.d. with variance v. Then

$$a_n = \frac{a_1}{mna_1 + 1} = \frac{1}{m}\left(\frac{1}{n + \frac{v}{h(\theta_1)^2}}\right). \tag{5}$$

Comparing the asymptotically optimal step-size—which is $\frac{1}{m}n^{-1}$ in this case—with (5), we learn that

- They are equivalent asymptotically.
- For all n, $a_n^* \geq a_n$, and $|a_n - a_n^*|$ is large when v is large.

Therefore, the larger the noise, the further a_n^* is away from the optimum during the finite horizon. In other words, there is still room for improvement in terms of the design of step-size. A simulation result which will be presented later illustrates this point. We refer to the step-size derived above as the finite horizon deterministic step-size since it was obtained by solving a finite horizon DP problem.

2.2 Adaptive Step-sizes

2.2.1 Review of Some Adaptive Step-sizes

Several adaptive step-sizes have been proposed to improve the performance of the SA algorithms. The idea behind them is to utilize the information gathered during the progress of the algorithms. Most of the step-sizes described below can be applied to vector parameter cases. Only the simplified versions for scalar case will be discussed in the following. Detailed discussions can be found in [6, 8].

Venter [9] proposed an adaptive step-size which uses an estimate of $h'(\theta^*)$ to implement a_n^*. As long as the estimate converges, the step-size is shown to minimize the variance of the resulting limiting normal distribution [9, 5]. Kesten [10] proposed a different type of step-size which decreases only when a change of signs of $\hat{h}_{n+1}$ occurs. It can be described as

$$a_1 = 1,$$

$$a_{n+1} = \begin{cases} a_n & \text{if } \operatorname{sgn}(\hat{h}_{n+2}) = \operatorname{sgn}(\hat{h}_{n+1}), \\ \frac{1}{\frac{1}{a_n}+1} & \text{otherwise.} \end{cases}$$

The a.s. convergence was shown in [10].

A more complicated step-size based on a similar idea was formulated by Mirozahmedov and Uriasev [11] as

$$a_{n+1} = a_n \exp[\rho a_n \hat{h}_{n+2} \hat{h}_{n+1} - \delta a_n],$$

where ρ and δ are fixed constants. This step-size increases when the product of the successive estimates is positive and decreases otherwise. The term $-\delta a_n$ is added to guarantee the convergence of a_n to zero. The a.s. convergence of this algorithm is established in [11].

Another step-size, proposed by Pflug [8], measures the progress of the parameters:

$$a_{n+1} = \begin{cases} \alpha_1 a_n, & \text{if } \frac{|\theta_n - \theta_{n-M_n}|}{\sum_{i=n-M_n+1}^{n} |\theta_i - \theta_{i-1}|} \leq \alpha_2, \\ a_n, & \text{otherwise,} \end{cases}$$

where $\alpha_1 < 1$ and $\alpha_2 > 0$ are design constant. $\{M_n\}$ is an increasing sequence of nonnegative integers. Several adaptive step-sizes which use sequential statistical test to decide whether the step-size should be decreased or not are also discussed in [8]. These schemes approximate the distribution of θ_n by the limiting normal distribution and use it to estimate $E(\hat{h}_{n+1}\hat{h}_n)$. The estimate is then used to define a sequence of stopping times for decreasing the step-size.

All of the adaptive step-sizes discussed above are derived or designed based on the asymptotic behavior of the algorithms. On the contrary, we consider the problem from a finite horizon point of view.

2.2.2 Finite Horizon Step-size

Similar to the previous section, we may attempt to derive an adaptive step-size by solving a finite-horizon DP problem. Suppose a_n is allowed to be a random variable measurable with respect to $\mathcal{F}_n$. The objective is to choose step-size $\{a_n : 1 \leq n \leq N\}$ to minimize V_N. This problem is virtually a *Stochastic Dynamic Programming* problem [7] if we assume that the derivative function $h(\cdot)$ is known. The corresponding DP problem can be described as follows. The state equation is given by

$$\begin{aligned} z_{n+1} &= (1 - a_n m) z_n - m a_n \varepsilon_{n+1}, \\ z_n &\triangleq h(\theta_n), \end{aligned}$$

where a_n is the control input we wish to design. Consider the class of control laws that consist of a sequence of functions $\pi = \{\mu_1, ..., \mu_{N-1}\}$, where μ_n maps z_n into control $a_n = \mu_n(z_n)$. Given $z_1 = h(\theta_1)$, we wish to find a control law π that minimizes

$$J_\pi = E\{z_N^2\}$$

subject to the state equation.

Again, we try to solve the DP problem by the *Optimal Principle.* Let $J_k(z_k)$ be the optimal cost for the last $N - k$ period. In our case, $J_k(z)$ is the optimal cost we can achieve at step N if $z_k = z$. We assume $z_n \neq 0$ for $n \leq N$, then for the $N - 1$ period,

$$\begin{aligned} J_N &= z_N^2, \\ J_{N-1}(z_{N-1}) &= \min_{a_{N-1}} E_{\mathcal{F}_{N-1}}\{J_N(z_{N-1}, a_{N-1}, \varepsilon_N)\} \\ &= \min_{a_{N-1}} \{(1 - a_{N-1} m)^2 z_{N-1}^2 + m^2 a_{N-1}^2 v_N\} \\ &= \frac{v_N}{1 + \frac{v_N}{z_{N-1}^2}} \end{aligned}$$

with

$$a_{N-1} = \frac{z_{N-1}^2}{m(z_{N-1}^2 + v_N)}.$$

For N-2 period,

$$\begin{aligned} J_{N-2}(z_{N-2}) &= \min_{a_{N-2}} E_{\mathcal{F}_{N-2}}\{J_{N-1}(z_{N-2}, a_{N-2}, \varepsilon_{N-1})\} \\ &= \min_{a_{N-2}} E_{\mathcal{F}_{N-2}}\{\frac{v_N}{1 + \frac{v_N}{z_{N-1}^2}}\}. \end{aligned}$$

Unfortunately, since we don't know the exact distribution of $\{\varepsilon_n\}$, it is difficult to evaluate the above expectation and derive a closed-form solution for the DP problem. A closed-form solution was obtained only when $N = 2$. The optimal step-size corresponding to this case is

$$a_n = \frac{h(\theta_n)^2}{m[h(\theta_n)^2 + v_{n+1}]}. \tag{6}$$

We will refer to this step-size as the "one-step adaptive step-size".

Figure 1 shows the averaged square errors between θ_n and θ^* over 100 computer simulations for the SA algorithms with three different step-sizes: the asymptotically optimal step-size a_n^*, the finite horizon deterministic step-size and the adaptive step-size just derived. The linear derivative function used in the simulations is $h(\theta) = 0.1\theta - 0.5$, the noise $\{\varepsilon_{n+1}\}$ are i.i.d. and uniformly distributed on $[-5, 5]$. The latter two step-sizes did perform better than a_n^* over finite time in the simulation. This supports our motivation for finite horizon designs. One point that needs to be addressed is that we implemented the step-sizes in the simulation by assuming the necessary information is available. This is seldom true in practice.

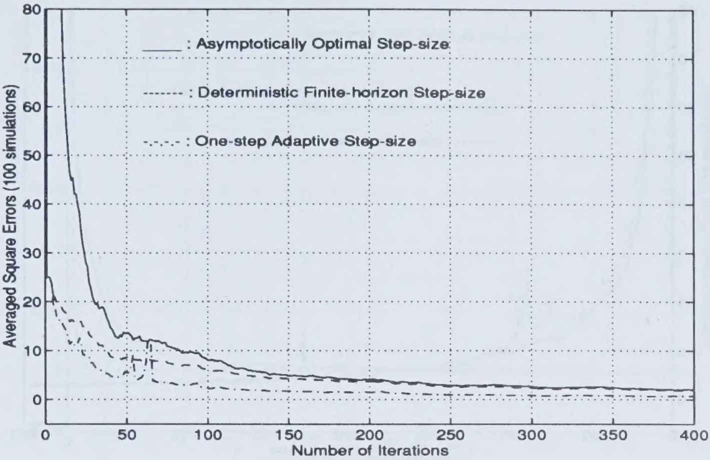


Figure 1: Averaged sample square error for three step-sizes.

2.3 Implementation Issues

As mentioned at the end of the last section, the step-sizes presented in the previous sections are not feasible in practice. Both step-sizes are derived assuming that prior information about the function and noise is available. However, this is not true for the problems that SA algorithms are meant to deal with. Hence, in order to implement these step-sizes, suitable estimation techniques need to be adopted to estimate the necessary information. We propose an implementation of (6) based on the *recursive least square* (RLS) technique [12] for the case of i.i.d. noise.

The basic idea is to use RLS to approximate the derivative function $h(\cdot)$ based on the noisy observations at θ_n's and then estimate the necessary information from the constructed function. Suppose (m_n, b_n) is the least square estimate of (m, b) based on the observations available at n, $\{(\theta_k, \hat{h}_{k+1}), 1 \leq k \leq n\}$, and define

$$\begin{aligned}
\tilde{h}_{n+1} &\triangleq m_n\theta_n + b_n, \\
\tilde{v}_{n+1} &\triangleq \frac{1}{n}\sum_{k=1}^{n}(\hat{h}_{k+1} - \tilde{h}_{k+1})^2.
\end{aligned}$$

Then we implement (6) by the following expression

$$\tilde{a}_n = \frac{\tilde{h}_{n+1}}{m_n[\tilde{h}_{n+1} + \tilde{v}_{n+1}]}.$$

The RLS recursions are

$$\begin{aligned}
\xi_{n+1} &= \xi_n + \frac{P_{n-1}\phi_n}{1 + \phi_n^T P_{n-1}\phi_n}[\hat{h}_{n+1} - \phi_n^T\xi_n], \\
P_n &= P_{n-1} - \frac{P_{n-1}\phi_n\phi_n^T P_{n-1}}{1 + \phi_n^T P_{n-1}\phi_n},
\end{aligned}$$

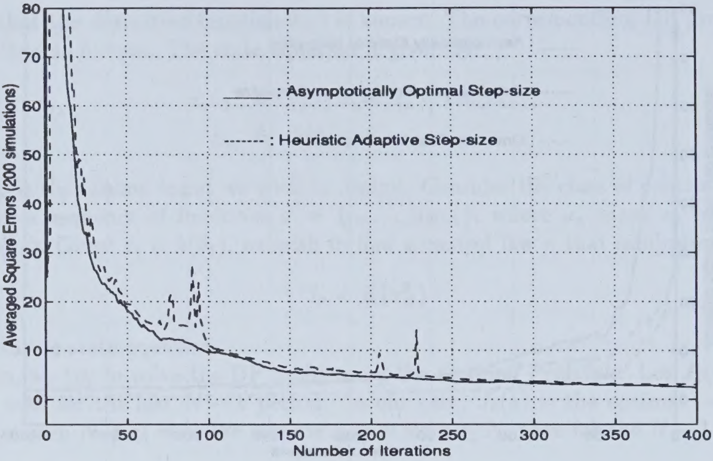


Figure 2: Averaged sample square errors for a_n^* and $\tilde{a}_n$.

$$\begin{aligned}\xi_n &\triangleq [m_n, b_n]^T, \\ \phi_n &\triangleq [\theta_n, 1]^T,\end{aligned}$$

with $\xi_1 = [\theta_1, 1]^T$ given and P_0 is any positive definite 2×2 matrix.

Figure 2 shows a simulation plot of the averaged square errors for the asymptotically optimal step-size a_n^* and the heuristic adaptive step-size $\tilde{a}_n$, applying to the same derivative function as in Figure 1. Unlike its original form (6), $\tilde{a}_n$ does not result in strictly better performance than a_n^*—the performance seems comparable. However, the implementation of $\tilde{a}_n$ does not require any prior information about $h(\cdot)$ but knowledge of $h'(\theta^*)$ is necessary for a_n^*.

3 Optimization of M/M/1 Queue

3.1 A Heuristic Step-size for General Derivative Functions

Based on the heuristic step-size derived for linear derivative functions, we can derive a heuristic step-size for the general case. The intuition behind this heuristic is to approximate the derivative function by a linear surface *locally* and then apply the heuristic step-size derived for linear functions. Specifically, we use RLS with *finite data window* [12] to locally construct a linear surface and then implement the heuristic step-size as in Section 2.3. To further improve the performance of the step-size, we multiply it by a decreasing sequence. The resulting heuristic step-size is applied to the optimization of a simple queue to illustrate it performance.

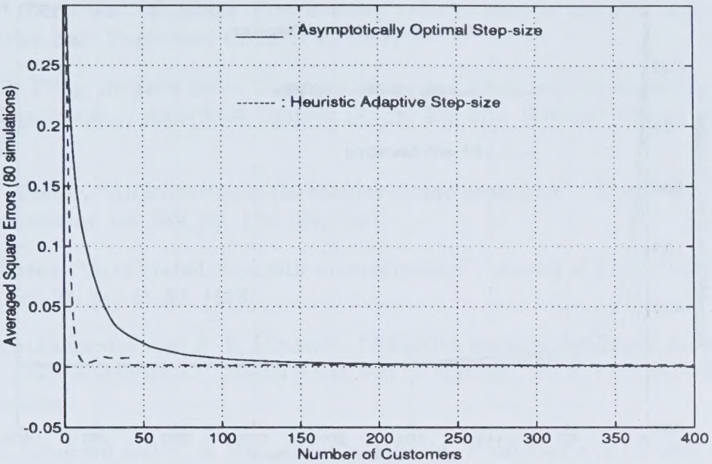


Figure 3: Performance of SA's with a_n^* and the heuristic adaptive step-size

3.2 Formulation and Assumptions

Optimizations of queues by IPA-based SA algorithms have been studied by several researchers [13, 14, 15, 16, 17]. We consider the optimization of a simple M/M/1 queue with fixed arrival rate λ and adjustable mean service time $\theta \in D$. The performance measure for this experiment is $J(\theta) = T(\theta) + C(\theta)$, where $T(\theta)$ is the steady state mean system time, and $C(\theta) = \frac{c}{\theta}, c > 0$. The problem is to minimize $J(\theta)$ with respect to θ. IPA algorithm presented in [15] is applied to estimate $\frac{dT}{d\theta}(\theta_n)$, which we denote by $\hat{g}_{n+1}$, and use $\hat{g}_{n+1} + \frac{dC}{d\theta}(\theta_n)$ as the estimate of $h(\theta_n)$. The SA algorithm is then of the form

$$\theta_{n+1} = \pi_{n+1}[\theta_n - a_n \hat{h}_{n+1}] = \pi_{n+1}[\theta_n - a_n(\hat{g}_{n+1} + \frac{dC}{d\theta}(\theta_n))].$$

In this simple case, we have analytical expressions for J and $h(\theta)$

$$\begin{aligned} J(\theta) &= \frac{\theta}{1-\lambda\theta} + \frac{c}{\theta}, \\ h(\theta) &= \frac{1}{(1-\lambda\theta)^2} - \frac{c}{\theta^2}. \end{aligned}$$

We test the heuristic step-size presented in the last section on this problem by computer simulations. Comparisons with several other SA algorithms are given next.

3.3 Simulation Results

In all the computer simulations, $\lambda = 1, \theta \in D = [0.0032, 0.952]$ and $c = 0.0625$. Hence $\theta^* = 0.2, h'(\theta^*) = 19.53125$ in this case. Figure 3 compares the heuristic adaptive step-size with the asymptotically optimal step-size a_n^*. Simulations shows dramatic gain in finite-time performance. The presented heuristic step-size is compared with the averaged SA algorithm [18], which was proposed for acceleration, in Figure 4. Our step-size performs better than the averaging scheme in this example.

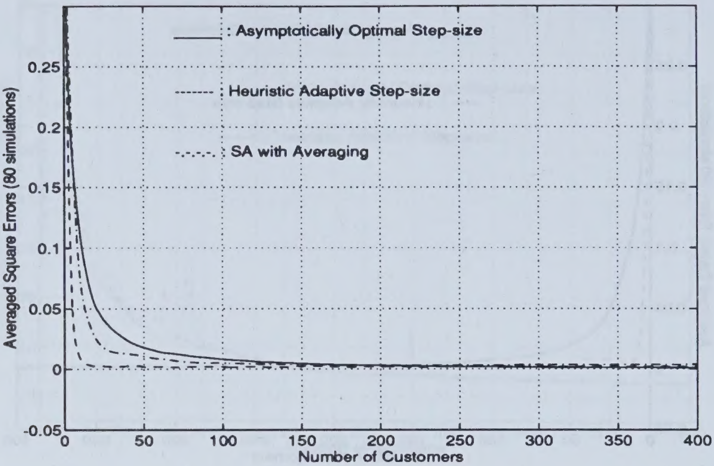


Figure 4: Performance of a_n^*, the heuristic adaptive step-size and the SA with averaging

4 Conclusions

Computer simulations did show signs of improvement by designing SA algorithm based on finite horizon criterion. However, no theoretical verification of the performance of the heuristic step-size is reached. In order to evaluate the finite-time performance of SA algorithms theoretically, a genuine rigorous technique remains to be developed.

References

[1] H. Robbins and S. Monroe, "A stochastic approximation method", *Annals of Mathematical Statistics*, vol. 22, pp. 400–407, 1951.

[2] H. K. Kushner and D. S. Clarke, *Stochastic Approximation Methods for Constrained and Unconstrained Systems*, Springer, New York, 1978.

[3] M. Metivier and P. Priouret, "Applications of a kushner and clark lemma to general classes of stochastic algorithms", *IEEE Trans. Inform. Theory*, vol. IT-30, pp. 140–151, Mar. 1984.

[4] K. L. Chung, "On a stochastic approximation method", *Annals of Mathematical Statistics*, vol. 25, pp. 463–483, 1954.

[5] V.Fabian, "On asymptotic normality in stochastic approximation", *Annals of Mathematical Statistics*, vol. 39, pp. 1327–1332, 1968.

[6] L. Goldstein, "On the choice of step size in the robbins-monro procedure", *Statistics and Probability Letters*, vol. 6, pp. 299–303, Apr. 1988.

[7] D. P. Bertsekas, *Dynamic Programming : Deterministic and Stochastic Models*, Prentice-Hall, Englewood Cliffs, N.J., 1987.

[8] G. Ch. Pflug, *Stepsize Rules, Stopping Times and Their Implementation in Stochastic Quasigradient Algorthms*, chapter 17, pp. 353–372, Springer-Verlag, New York, 1988.

[9] J. H. Venter, "An extention of the robbins-monro procedure", *Annals of Mathematical Statistics*, vol. 389, pp. 181–190, 1967.

[10] H. Kesten, "Accelerated stochastic approximation", *Annals of Mathematical Statistics*, vol. 29, pp. 41–59, 1958.

[11] F. Mirozahmedov and S. P. Uryasev, "Adaptive stepsize regulation for stochastic optimization algorithm", *Zurnal vicisl. mat. i. mat. fiz.*, vol. 6, pp. 1314–1325, 1983, In Russian.

[12] G. C. Goodwin and K. S. Sin, *Adaptive Filtering Prediction and Control*, Prentice-Hall, Englewood Cliffs, New Jersey, 1984.

[13] M. C. Fu, "Convergence of a stochastic approximation algorithm for the GI/G/1 queue using infinitesimal perturbation analysis", *J. Optim. Theory Appl.*, vol. 65, pp. 149–160, 1990.

[14] E. K. P. Chong and P.J. Ramadge, "Convergence of recursive optimization algorithms using infintesimal perturbation analysis estimates", *Discrete Event Dynamic Systems: Theory and Applications*, vol. , pp. 339–372, 1992.

[15] E. K. P. Chong and P. J. Ramadge, "Optimization of queues using an infinitesimal perturbation analysis-based stochastic algorithm with general update times", *SIAM J. Contr. Opt.*, vol. 31, pp. 698–732, May 1993.

[16] P. L'ecuyer, N. Giroux, and P. W. Glynn, "Stochastic optimization by simulation: Numerical experiments with the M/M/1 queue in steady-state", Manuscript, June 1993.

[17] P. L'ecuyer and P. W. Glynn, "Stochastic optimization by simulation: Convergence proofs for the GI/G/1 queue in steady-state", Manuscript, June 1993.

[18] B. T. Polyak and A. B. Juditsky, "Acceleration of stochastic approximation by averaging", *SIAM J. Contr. Opt.*, vol. 30, pp. 838–855, July 1992.

Likelihood Ratio Methods via Conditional Monte Carlo and Splitting

Michael C. Fu† Jian-Qiang Hu‡ Leyuan Shi†
†University of Maryland, College Park ‡Boston University

Abstract

For the problem of steady-state gradient estimation for discrete-time Markov chains, we propose a variance reduction technique for the likelihood ratio method, which in its original form suffers from the problem of increasing variance as the horizon is increased. Among the numerous proposals to alleviate the problem is the use of conditional expectation, which we apply along with splitting in a novel way to derive estimators which also provably reduce variance compared with the original estimators. Furthermore, we show an interesting connection between the resulting estimators and smoothed perturbation analysis estimators, which are also derived using conditioning arguments.

1 Introduction

We consider the problem of steady-state gradient estimation for discrete-time Markov chains via the likelihood ratio (LR) method. It is well-known that the "naive" form of the LR estimator suffers from the problem of unbounded (linear growth) variance as the horizon increases, and so the estimator is useless for steady-state estimation. Under regenerative conditions, the estimator can be easily modified to alleviate this problem (cf. Glynn 1987), but again the magnitude of the variance may be large for systems with long regenerative cycles. Other different proposals to reduce variance include the work of Zhang and Ho (1991), which doesn't require regeneration, and techniques using some form of conditioning, e.g., Fox and Glasserman (1991).

We develop an estimator based on ideas similar to the latter work, by applying conditional expectation along with splitting in a novel way to derive estimators which also have bounded variance. The splitting idea involves a "cut-and-paste" idea which takes advantage of the fact that we are interested in steady-state performance. The resulting estimator can be viewed as an example where variance is provably reduced (over the original estimator) by conditioning. We then establish regenerative structure in the resulting estimator in order to prove strong consistency.

In addition, the estimator contains some interesting connections to smoothed perturbation analysis (SPA) estimators, which also used conditional expectation, and to Poisson's equation. The work of Fox and Glasserman (1991) also involved Poisson's equation, but in a completely different manner. Their work uses Poisson's equation as a starting point for deriving their estimators, whereas our connection with Poisson's equation is its use in estimating certain terms in our estimator.

The rest of the paper is organized as follows. In Section 2, we derive an estimator via conditional expectation and splitting for the finite horizon case, and discuss the connection of the resulting estimator to SPA estimators. In Section 3, we prove strong consistency of the estimator by establishing regenerative structure for the estimator. Exploiting the fact that we are interested in steady-state estimation, we derive a simplified estimator in Section 4, and discuss briefly the connection with Poisson's equation.

2 Derivation of the Estimator

Let $\{X_n\}$ be a finite-state, irreducible and positive recurrent, discrete-time Markov chain, with state space $\mathbf{S}$ and (stationary) transition matrix $\{P_\theta(i,j) : i,j \in \mathbf{S}\}$, where $\theta \in \Theta$ is a parameter of interest and Θ is an open set. Our results also extend easily to the countable state space case, with the assumption that appropriate (possibly infinite) sums converge.

We consider performance measures which are additive functionals of the form

$$f_n = \frac{1}{n}\sum_{i=0}^{n} g(X_i),$$

where $g(\cdot)$ is a mapping: $\mathbf{S} \to \Re$, independent of θ. We will also assume that the initial distribution is independent of θ. Cases where θ is in the mapping $g(\cdot)$ and in the initial distribution cause no additional difficulties (see e.g., Glynn 1987), in the sense that the variance problem we will be addressing is fully captured by the parameter being in the transition probabilities.

In particular, we will be interested in the steady-state version of performance taken as the limiting value:

$$\bar{f} = \lim_{n\to\infty} f_n = \lim_{n\to\infty} \frac{1}{n}\sum_{i=0}^{n} g(X_i).$$

We wish to estimate $d\bar{f}/d\theta$ using the likelihood ratio (LR) method. The straightforward finite horizon LR estimator given by

$$\frac{df_n}{d\theta} = \frac{1}{n}\sum_{i=0}^{n} g(X_i)\sum_{j=0}^{n-1}\frac{dP_\theta(X_j,X_{j+1})/d\theta}{P_\theta(X_j,X_{j+1})} \tag{1}$$

is unbiased, but has the undersirable property of linear variance growth with n, making it inapplicable in the steady-state setting. One way is reduce variance to is to use the regenerative method, as has been shown by Glynn (1987) and Reiman and Weiss (1986). Another completely different method has been proposed by Zhang and Ho (1991). In this paper, we propose a method based on (extended) conditional expectation and splitting.

Using

$$E_\theta\left[\left.\frac{P_{\theta'}(X_j,X_{j+1})}{P_\theta(X_j,X_{j+1})}\right| X_0,\ldots,X_j\right] = 1,$$

we have

$$E\left[\left.\frac{dP_\theta(X_j,X_{j+1})/d\theta}{P_\theta(X_j,X_{j+1})}\right| X_0,\ldots,X_j\right] = 0. \tag{2}$$

Thus, taking conditional expectation in (1), gives

$$\begin{aligned}
& E\left[\frac{1}{n}\sum_{i=0}^{n} g(X_i)\sum_{j=0}^{n-1}\frac{dP_\theta(X_j,X_{j+1})/d\theta}{P_\theta(X_j,X_{j+1})}\,\middle|\, X_0,\ldots,X_j\right] \\
&= \frac{1}{n}\sum_{j=0}^{n-1} E\left[\sum_{i=0}^{n} g(X_i)\frac{dP_\theta(X_j,X_{j+1})/d\theta}{P_\theta(X_j,X_{j+1})}\,\middle|\, X_0,\ldots,X_j\right] \\
&= \frac{1}{n}\sum_{j=0}^{n-1} E\left[\sum_{i=j+1}^{n} g(X_i)\frac{dP_\theta(X_j,X_{j+1})/d\theta}{P_\theta(X_j,X_{j+1})}\,\middle|\, X_0,\ldots,X_j\right],
\end{aligned} \tag{3}$$

the last equality following from (2). The same conditioning was used by Glynn (1987) to reduce variance of LR estimators in a regenerative setting.

Let $\tilde{x} \in \mathbf{S}$ be an arbitrary but fixed state. Since $\sum_{x\in\mathbf{S}} P_\theta(X_j, x) = 1$ (given X_j), we have

$$dP_\theta(X_j,\tilde{x})/d\theta = -\sum_{x\in\mathbf{S},x\neq\tilde{x}} dP_\theta(X_j,x)/d\theta \text{ (given } X_j). \tag{4}$$

Using this relationship, we go one step further by conditioning on X_{j+1} (the previous conditioning on $X_0, \ldots, X_j$ understood but not explicitly written here):

$$\begin{aligned}
& E\left[\sum_{i=j+1}^{n} g(X_i)\frac{dP_\theta(X_j,X_{j+1})/d\theta}{P_\theta(X_j,X_{j+1})}\right] \\
&= \sum_{x\in\mathbf{S}}\frac{dP_\theta(X_j,x)}{d\theta} E\left[\sum_{i=j+1}^{n} g(X_i)\,\middle|\, X_{j+1}=x\right] \\
&= \sum_{x\in\mathbf{S},x\neq\tilde{x}}\frac{dP_\theta(X_j,x)}{d\theta}\left(E\left[\sum_{i=j+1}^{n} g(X_i)\,\middle|\, X_{j+1}=x\right] - E\left[\sum_{i=j+1}^{n} g(X_i)\,\middle|\, X_{j+1}=\tilde{x}\right]\right)
\end{aligned} \tag{5}$$

the latter following from application of (4).

Thus, substituting (5) into (3), our full estimator becomes

$$\frac{1}{n}\sum_{j=0}^{n-1}\sum_{x\in\mathbf{S},x\neq\tilde{x}}\frac{dP_\theta(X_j,x)}{d\theta}\left(E\left[\sum_{i=j+1}^{n} g(X_i)\,\middle|\, X_{j+1}=x\right] - E\left[\sum_{i=j+1}^{n} g(X_i)\,\middle|\, X_{j+1}=\tilde{x}\right]\right). \tag{6}$$

Note that the difference term

$$\left(\sum_{i=j+1}^{n} g(X_i)\,\middle|\, X_{j+1}=x\right) - \left(\sum_{i=j+1}^{n} g(X_i)\,\middle|\, X_{j+1}=\tilde{x}\right) \tag{7}$$

is a difference in the performance measure defined on two different sample paths, one with initial state $\tilde{x}$ and the other with initial state $x \neq \tilde{x}$. As it stands, the straightforward estimator for this term still clearly leads to linear variance growth. We rectify this problem by a splitting construction, where the sample path with initial state $\tilde{x}$ is constructed from the sample path with initial state $x \neq \tilde{x}$. It is interesting to note that (6) turns out to be the same as the estimator derived using the method of Discontinuous Perturbation Analysis (DPA) (see Shi 1992).

We denote the first hitting time to $\tilde{x}$ from $X_j = x$ by

$$m_j(x) = \min\{m : X_m = \tilde{x} | X_j = x, m \geq j\}.$$

For any $x \in \mathbf{S}$, the two chains

$$\{X_i | X_{j+1} = x, i \geq m_{j+1}(x)\} \text{ and } \{X_i | X_{j+1} = \tilde{x}, i \geq j+1\}$$

are equal in distribution, so our construction is simply to use the sample path from $m_{j+1}(x)$ onwards in $\{X_i | X_{j+1} = x\}$ to estimate the quantities defined on $\{X_i | X_{j+1} = \tilde{x}\}$. Thus, (7) is equal in expectation to

$$\begin{aligned} &\left(\sum_{i=j+1}^{n} g(X_i) \middle| X_{j+1} = x\right) - \left(\sum_{i=m_{j+1}(x)}^{n+m_{j+1}(x)-j-1} g(X_i) \middle| X_{j+1} = x\right) \\ &= \left(\sum_{i=j+1}^{\min(m_{j+1}(x)-1,n)} g(X_i) - \sum_{i=\max(m_{j+1}(x),n+1)}^{n+m_{j+1}(x)-j-1} g(X_i)\right)\Bigg| X_{j+1} = x. \end{aligned} \tag{8}$$

The splitting construction leading to (8) is shown in Figure 1. The important thing to note is that the term no longer has increasing variance with the horizon length, but instead depends on the hitting times $m_{j+1}(x)$. Finally, the full estimator given by (6) is equal in expectation to

$$\begin{aligned} &\frac{1}{n}\sum_{j=0}^{n-1}\sum_{x\in\mathbf{S},x\neq\tilde{x}} \frac{dP_\theta(X_j,x)}{d\theta} E\left[\left(\sum_{i=j+1}^{\min(m_{j+1}(x)-1,n)} g(X_i) - \sum_{i=\max(m_{j+1}(x),n+1)}^{n+m_{j+1}(x)-j-1} g(X_i)\right)\Bigg| X_{j+1} = x\right] \\ &= \frac{1}{n}\sum_{(x',x)\in\mathbf{S}\times\mathbf{S}\backslash\{\tilde{x}\}} \frac{dP_\theta(x',x)}{d\theta}\sum_{j=0}^{n-1} I\{X_j = x', X_{j+1} = x\} \\ &\quad E\left[\sum_{i=j+1}^{\min(m_{j+1}(x)-1,n)} g(X_i) - \sum_{i=\max(m_{j+1}(x),n+1)}^{n+m_{j+1}(x)-j-1} g(X_i)\right] \\ &= \frac{1}{n}\sum_{(x',x)\in\mathbf{S}\times\mathbf{S}\backslash\{\tilde{x}\}} \frac{dP_\theta(x',x)}{d\theta}\sum_{j\in J_n(x',x)} E\left[\sum_{i=j+1}^{\min(m_{j+1}(x)-1,n)} g(X_i) - \sum_{i=\max(m_{j+1}(x),n+1)}^{n+m_{j+1}(x)-j-1} g(X_i)\right], \end{aligned} \tag{9}$$

where $I\{\cdot\}$ is the indicator function, and to simplify notation, we have defined the set $J_n(x', x) = \{j : X_j = x', X_{j+1} = x, 0 \leq j \leq n-1\}$. We emphasize that the expectations in (9) are taken under the condition $X_{j+1} = x$, indicated by the summation over $J_n(x', x)$. Also note that although $\tilde{x}$ no longer explicitly appears in the inner term, the definition of $m_{j+1}(x)$ is based on the choice of $\tilde{x}$.

We now use (1) to derive another estimator different from (6). Since

$$\begin{aligned} &E\left[\left(\sum_{i=j+1}^{n} g(X_i) \middle| X_{j+1} = \tilde{x}\right)\left(\frac{dP_\theta(X_j, X_{j+1})/d\theta}{P_\theta(X_j, X_{j+1})}\right)\right] \\ &= E\left[\sum_{i=j+1}^{n} g(X_i) \middle| X_{j+1} = \tilde{x}\right] E\left[\frac{dP_\theta(X_j, X_{j+1})/d\theta}{P_\theta(X_j, X_{j+1})}\right] = 0, \end{aligned}$$

we have

$$\begin{aligned}
&\frac{1}{n}\sum_{j=0}^{n-1} E\left[\sum_{i=j+1}^{n} g(X_i)\frac{dP_\theta(X_j,X_{j+1})/d\theta}{P_\theta(X_j,X_{j+1})}\right] \\
&= \frac{1}{n}\sum_{j=0}^{n-1} E\left[\sum_{i=j+1}^{n} g(X_i)\frac{dP_\theta(X_j,X_{j+1})/d\theta}{P_\theta(X_j,X_{j+1})}\right] \\
&- \frac{1}{n}\sum_{j=0}^{n-1} E\left[\left(\sum_{i=j+1}^{n} g(X_i)\,\middle|\, X_{j+1}=\tilde{x}\right)\left(\frac{dP_\theta(X_j,X_{j+1})/d\theta}{P_\theta(X_j,X_{j+1})}\right)\right] \\
&= \frac{1}{n}\sum_{j=0}^{n-1} E\left[\left(\sum_{i=j+1}^{n} g(X_i)-\left(\sum_{i=j+1}^{n} g(X_i)\,\middle|\, X_{j+1}=\tilde{x}\right)\right)\frac{dP_\theta(X_j,X_{j+1})/d\theta}{P_\theta(X_j,X_{j+1})}\right]. \qquad (10)
\end{aligned}$$

Again, the difference term

$$\left(\sum_{i=j+1}^{n} g(X_i)-\left(\sum_{i=j+1}^{n} g(X_i)\,\middle|\, X_{j+1}=\tilde{x}\right)\right) \qquad (11)$$

is a difference in the performance measure defined on two different sample paths, the original sample path and a sample path with initial state $X_{j+1}=\tilde{x}$. It is clear that the estimator given by (10) is simpler than the one given by (6) since only one difference needs to be calculated at each state in (10). It is also interesting to note that estimators (6) and (10) are very similar to the smoothed perturbation analysis (SPA) estimators derived in Fu and Hu (1992). In fact, (10) becomes an SPA estimator for some special cases. For example, if $P(1,2)=\theta$, $P(1,3)=1-\sum_{x\in\mathbf{S},x\neq 3})$, and the other $P(i,j)$'s are independent of θ, then by choosing $\tilde{x}=3$, (10) is equal (in expectation) to

$$\frac{1}{n}\sum_{X_j=1,X_{j+1}=2}\left(\left(\sum_{i=j+1}^{n} g(X_i)\,\middle|\, X_{j+1}=2\right)-\left(\sum_{i=j+1}^{n} g(X_i)\,\middle|\, X_{j+1}=3\right)\right)\frac{dP_\theta(1,2)/d\theta}{P_\theta(1,2)},$$

which is the corresponding SPA estimator.

Furthermore, by conditioning on X_{j+1}, we get

$$\begin{aligned}
&E\left[\frac{1}{n}\sum_{j=0}^{n-1} E_{X_{j+1}}\left[\left(\sum_{i=j+1}^{n} g(X_i)-\left(\sum_{i=j+1}^{n} g(X_i)\,\middle|\, X_{j+1}=\tilde{x}\right)\right)\frac{dP_\theta(X_j,X_{j+1})/d\theta}{P_\theta(X_j,X_{j+1})}\right]\right] \\
&= \frac{1}{n}\sum_{j=0}^{n-1}\sum_{x\in\mathbf{S},x\neq\tilde{x}}\frac{dP_\theta(X_j,x)}{d\theta}\left(E\left[\sum_{i=j+1}^{n} g(X_i)\,\middle|\, X_{j+1}=x\right]-E\left[\sum_{i=j+1}^{n} g(X_i)\,\middle|\, X_{j+1}=\tilde{x}\right]\right),
\end{aligned}$$

i.e., (6) is the (extended) conditional expectation of (10).

Before closing this section we want to point out the fact that the estimator as defined by (6) involves quantities (the conditional expectations) which have to be estimated separately by generating additional sample paths with given initial states. By modifying it to (9), we make use of subpaths of the original sample path to estimate these quantities and thus reduce such additional sample paths.

3 Strong Consistency

Since we are really interested in $d\bar{f}/d\theta$, we need the estimator (9) to a.s. converge to $d\bar{f}/d\theta$ as $n \to \infty$. We follow the approach taken by Fu and Hu (1993) in proving strong consistency for a smoothed perturbation analysis estimator. First, we have to make sure that the a.s. limit of the estimator exists. One of the most common conditions used to prove the existence of the a.s. limit is the regenerative condition. Moreover, the regenerative condition together with the unbiasedness condition also guarantees that the limit is equal to $d\bar{f}/d\theta$. The estimator (6) does not have regenerative structure, which is another reason why we modified it to obtain (9), for which we now establish regenerative structure.

We first focus on one term of (9)

$$\sum_{j\in J_n} E\left[\sum_{i=j+1}^{\min(m_{j+1}-1,n)} g(X_i) - \sum_{i=\max(m_{j+1},n+1)}^{n+m_{j+1}-j-1} g(X_i)\right], \tag{12}$$

where we have suppressed arguments x' and x in J_n and m_{j+1} to lighten some of the notational burden. Defining

$$\begin{aligned} G_n &= \sum_{j\in J_{n+1}} E\left[\sum_{i=j+1}^{\min(m_{j+1}-1,n+1)} g(X_i) - \sum_{i=\max(m_{j+1},n+2)}^{n+m_{j+1}-j} g(X_i)\right] \\ &\quad - \sum_{j\in J_n} E\left[\sum_{i=j+1}^{\min(m_{j+1}-1,n)} g(X_i) - \sum_{i=\max(m_{j+1},n+1)}^{n+m_{j+1}-j-1} g(X_i)\right], \end{aligned} \tag{13}$$

we have

Lemma 1. $\{G_n, n = 1, 2, \ldots\}$ is a regenerative process.

Proof. Define $\xi_k = \min\{j : X_j = \tilde{x}, j > \xi_{k-1}\}$ $(k = 1, 2, \ldots)$, with $\xi_0 = -1$, i.e., ξ_k is the time of kth visit of the chain to state $\tilde{x}$. Since $\{X_n\}$ is a finite-state, irreducible and positive recurrent, Markov chain, $\{\xi_k, k = 1, 2, \ldots\}$ constitute a set of regenerative points for the chain. To complete our proof, we show that $\{\xi_k, k = 1, 2, \ldots\}$ also constitute a set of regenerative points for $\{G_n, n = 1, 2, \ldots\}$ by showing that $\{\sum_{j=\xi_k}^{\xi_{k+1}-1} G_j, k = 1, 2, \ldots\}$ are i.i.d.

By definition, we have

$$\begin{aligned} \sum_{j=\xi_k}^{\xi_{k+1}-1} G_j &= \sum_{j\in J_{\xi_{k+1}}} E\left[\sum_{i=j+1}^{\min(m_{j+1}-1,\xi_{k+1})} g(X_i) - \sum_{i=\max(m_{j+1},\xi_{k+1}+1)}^{\xi_{k+1}+m_{j+1}-j-1} g(X_i)\right] \\ &\quad - \sum_{j\in J_{\xi_k}} E\left[\sum_{i=j+1}^{\min(m_{j+1}-1,\xi_k)} g(X_i) - \sum_{i=\max(m_{j+1},\xi_k+1)}^{\xi_k+m_{j+1}-j-1} g(X_i)\right] \\ &= \sum_{j\in J_{\xi_{k+1}}\setminus J_{\xi_k}} E\left[\sum_{i=j+1}^{\min(m_{j+1}-1,\xi_{k+1})} g(X_i) - \sum_{i=\max(m_{j+1},\xi_{k+1}+1)}^{\xi_{k+1}+m_{j+1}-j-1} g(X_i)\right] \\ &\quad + \sum_{j\in J_{\xi_k}} E\Bigg[\Bigg(\sum_{i=j+1}^{\min(m_{j+1}-1,\xi_{k+1})} g(X_i) - \sum_{i=\max(m_{j+1},\xi_{k+1}+1)}^{\xi_{k+1}+m_{j+1}-j-1} g(X_i)\Bigg) \end{aligned}$$

$$- \left(\sum_{i=j+1}^{\min(m_{j+1}-1,\xi_k)} g(X_i) - \sum_{i=\max(m_{j+1},\xi_k+1)}^{\xi_k+m_{j+1}-j-1} g(X_i) \right) \Bigg], \tag{14}$$

the latter using $J_{\xi_k} \subset J_{\xi_{k+1}}$. Next, by the definition of ξ_k and m_{j+1} we have $m_{j+1} \leq \xi_k (< \xi_{k+1})$ for $j \in J_{\xi_k}$ and $m_{j+1} = \xi_{k+1}$ for $j \in J_{\xi_{k+1}} \backslash J_{\xi_k}$. Therefore, (14) can be further simplified as

$$\begin{aligned}
\sum_{j=\xi_k}^{\xi_{k+1}-1} G_j &= \sum_{j \in J_{\xi_{k+1}} \backslash J_{\xi_k}} E\left[\sum_{i=j+1}^{\xi_{k+1}-1} g(X_i) - \sum_{i=\xi_{k+1}+1}^{\xi_{k+1}+\xi_{k+1}-j-1} g(X_i) \right] \\
&\quad + \sum_{j \in J_{\xi_k}} E\left[\left(\sum_{i=j+1}^{m_{j+1}-1} g(X_i) - \sum_{i=\xi_{k+1}+1}^{\xi_{k+1}+m_{j+1}-j-1} g(X_i) \right) \right. \\
&\quad \left. - \left(\sum_{i=j+1}^{m_{j+1}-1} g(X_i) - \sum_{i=\xi_k+1}^{\xi_k+m_{j+1}-j-1} g(X_i) \right) \right] \\
&= \sum_{j \in J_{\xi_{k+1}} \backslash J_{\xi_k}} E\left[\sum_{i=j+1}^{\xi_{k+1}-1} g(X_i) - \sum_{i=\xi_{k+1}+1}^{\xi_{k+1}+\xi_{k+1}-j-1} g(X_i) \right] \\
&\quad + \sum_{j \in J_{\xi_k}} E\left[\sum_{i=\xi_k+1}^{\xi_k+m_{j+1}-j-1} g(X_i) - \sum_{i=\xi_{k+1}+1}^{\xi_{k+1}+m_{j+1}-j-1} g(X_i) \right]
\end{aligned} \tag{15}$$

Since $\{X_i, i \geq \xi_k + 1\}$ and $\{X_i, i \geq \xi_{k+1} + 1\}$ are equal in distribution, we have

$$E\left[\sum_{i=\xi_k+1}^{\xi_k+m_{j+1}-j-1} g(X_i) - \sum_{i=\xi_{k+1}+1}^{\xi_{k+1}+m_{j+1}-j-1} g(X_i) \right] = 0, \tag{16}$$

and

$$E\left[\sum_{i=\xi_{k+1}+1}^{\xi_{k+1}+\xi_{k+1}-j-1} g(X_i) \right] = E\left[\sum_{i=\xi_k+1}^{\xi_k+\xi_{k+1}-j-1} g(X_i) \right]. \tag{17}$$

It then follows from (15), (16), and (17) that

$$\sum_{j=\xi_k}^{\xi_{k+1}-1} G_j = \sum_{j \in J_{\xi_{k+1}} \backslash J_{\xi_k}} E\left[\sum_{i=j+1}^{\xi_{k+1}-1} g(X_i) - \sum_{i=\xi_k+1}^{\xi_k+\xi_{k+1}-j-1} g(X_i) \right], \tag{18}$$

which implies that $\{\sum_{j=\xi_k}^{\xi_{k+1}-1} G_j, k = 1, 2, \ldots\}$ are i.i.d., since for each k, the expression depends only on $\{X_i, \xi_k \leq i \leq \xi_{k+1} - 1\}$. □

If we define

$$G_{\Sigma,n} = \sum_{(x',x) \in \mathrm{S} \times \mathrm{S} \backslash \{\bar{x}\}} G_n, \tag{19}$$

then based on Lemma 1, it is not difficult for us to obtain

Lemma 1′. $\{G_{\Sigma,n}, n = 1, 2, \ldots\}$ is a regenerative process with regenerative points $\{\xi_k, k = 1, 2, \ldots\}$.

Rewriting (9) as

$$\frac{1}{n} \sum_{i=0}^{n} G_{\Sigma,i}, \tag{20}$$

we are now ready to state

Theorem 1. If $\{G_{\Sigma,n}, n = 1, 2, \ldots\}$ is a regenerative process with regenerative points $\{\xi_k, k = 1, 2, \ldots\}$, and $E[n^{-1}\sum_{i=0}^{n} G_{\Sigma,n}] = dE[f_n]/d\theta$, then if

(i) $E[\xi_{k+1} - \xi_k] < \infty$,

(ii) $E[\sum_{i=\xi_k}^{\xi_{k+1}-1} g(X_i)] < \infty$,

(iii) $E[\sum_{i=\xi_k}^{\xi_{k+1}-1} |G_{\Sigma,i}|] < \infty$,

(iv) $E[n^{-1}\sum_{i=1}^{n} |G_{\Sigma,i}|] \leq K \ \forall\ \theta \in \Theta$, $n = 1, 2, \ldots$, where K is some constant not depending on θ or n,

$$\text{then} \qquad \lim_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n} G_{\Sigma,i} = \frac{d\bar{f}}{d\theta} \qquad \text{a.e. on } \Theta.$$

Proof. Follows from Theorem 2 in Glasserman, Hu and Strickland (1992). □

Basically the theorem says that the estimator is strongly consistent for the steady-state performance measure if it has regenerative structure and is unbiased for the finite-horizon performance measure. The conditions (i)-(iv) are mainly technical in nature and can in general be established under some moments conditions on the length of a regenerative cycle and the average number of states in a regenerative cycle. In fact, most of these conditions are also required to guarantee the existence of the steady-state performance measure as well as the steady-state estimator. Henceforth, we shall assume they are in force.

4 The Limiting Value of the Estimator

In the last section we have shown that the estimator (9) a.s. converges to $d\bar{f}/d\theta$. As we pointed out before, all the quantities involved in (9) can be estimated based on one sample path of the Markov chain. However, it is not very efficient, since at each state a separate (splitting) quantity needs to be obtained. In this section, we shall further simplify it based on the fact that we are interested in large n (steady state).

Based on the regenerative structure we obtained in Lemma 1 and (18), we know that the a.s. limit of (9) as $n \to \infty$ is equal to

$$\frac{\sum_{(x',x)\in \mathbf{S}\times\mathbf{S}\setminus\{\tilde{x}\}} \frac{dP_\theta(x',x)}{d\theta} E\left[\sum_{j\in J_{\xi_{k+1}}\setminus J_{\xi_k}} E\left[\sum_{i=j+1}^{\xi_{k+1}-1} g(X_i) - \sum_{i=\xi_k+1}^{\xi_k+\xi_{k+1}-j-1} g(X_i)\right]\right]}{E[\xi_{k+1} - \xi_k]}. \tag{21}$$

Following the same derivation as we used in proving Lemma 1, we can also show that the estimator given by

$$\frac{1}{n} \sum_{(x',x)\in \mathbf{S}\times\mathbf{S}\setminus\{\tilde{x}\}} \frac{dP_\theta(x',x)}{d\theta} \sum_{j\in J_n(x',x)} E\left[\sum_{i=j+1}^{m_{j+1}(x)-1} g(X_i) - \sum_{i=n+1}^{n+m_{j+1}(x)-j-1} g(X_i)\right], \tag{22}$$

also a.s. converges to (21) as $n \to \infty$. Hence (9) and (22) both converge to $d\bar{f}/d\theta$ as $n \to \infty$.

For any $x \in \mathbf{S}$, we denote the chain starting from initial state $X_0 = x$ by superscript (x), i.e.,

$$\{X_n | X_0 = x\} = \{X_n^{(x)}\},$$

and use $m(x)$ to denote the first hitting time of the chain $\{X_n^{(x)}\}$ to $\tilde{x}$, i.e.,

$$m(x) = \min\{n : X_n^{(x)} = \tilde{x}, n \geq 0\}.$$

Then, (22) is equal in expectation to

$$\frac{1}{n} \sum_{(x',x)\in \mathbf{S}\times\mathbf{S}\setminus\{\tilde{x}\}} \frac{dP_\theta(x',x)}{d\theta} \sum_{j\in J_n(x',x)} E\left[\sum_{i=1}^{m(x)} g(X_i^{(x)}) - \sum_{i=n+1}^{n+m(x)} g(X_i)\right]. \tag{23}$$

We denote the limiting distribution of $\{X_n\}$ by $[p(x), x \in \mathbf{S}]$, where

$$p(x) = \lim_{n\to\infty} P\{X_n = x\},$$

and let $\{\tilde{X}_n\}$ be the stationary version of $\{X_n\}$, i.e., $\tilde{X}_n \sim [p(x), x \in \mathbf{S}]$ for all n. Then, we have for each $j \in J_n(x', x)$

$$\lim_{n\to\infty} E\left[\sum_{i=n+1}^{n+m(x)} g(X_i)\right] = E\left[\sum_{i=1}^{m(x)} g(\tilde{X}_i)\right],$$

where $\{\tilde{X}_n\}$ is independent of $m(x)$. Since by definition $E[g(\tilde{X}_n)] = \bar{f}$ for every n, we have

$$E\left[\sum_{i=1}^{m(x)} g(\tilde{X}_i)\right] = E[m(x)]\bar{f}.$$

Also note that

$$\lim_{n\to\infty} \frac{1}{n} \sum_{j\in J_n(x',x)} 1 = p(x')P_\theta(x', x).$$

Summarizing the above, we have

$$\begin{aligned}&\lim_{n\to\infty} \frac{1}{n} \sum_{(x',x)\in\mathbf{S}\times\mathbf{S}\setminus\{\tilde{x}\}} \frac{dP_\theta(x',x)}{d\theta} \sum_{j\in J_n(x',x)} E\left[\sum_{i=j+1}^{m_{j+1}(x)-1} g(X_i) - \sum_{i=n+1}^{n+m_{j+1}(x)-j-1} g(X_i)\right] \\ &= \sum_{x'\in\mathbf{S}} p(x') \sum_{x\in\mathbf{S}, x\neq\tilde{x}} \frac{dP_\theta(x',x)}{d\theta} P_\theta(x', x) \left(E\left[\sum_{i=1}^{m(x)} g(X_i^{(x)})\right] - E[m(x)]\bar{f}\right). \end{aligned} \tag{24}$$

It is clear that $P(x')$, $E\left[\sum_{i=1}^{m(x)} g(X_i^{(x)})\right]$, $E[m(x)]$, and $\bar{f}$ involved in (24) can be estimated based on the original sample path. In fact, $E\left[\sum_{i=1}^{m(x)} g(X_i^{(x)})\right]$ and $E[m(x)]$ are the key quantities to be estimated, where $E[m(x)]$ is the expected first hitting time to $\tilde{x}$ starting from state x, and $E\left[\sum_{i=1}^{m(x)} g(X_i^{(x)})\right]$ is the corresponding expected cumulative reward. They both satisfy a set of linear equations also known as Poisson's equation, where $\{\tilde{x}\}$ can be thought of as the set of absorbing states. The estimator also bears much resemblance to the SPA estimator in Fu and Hu (1991), where continuous time Markov chains were considered.

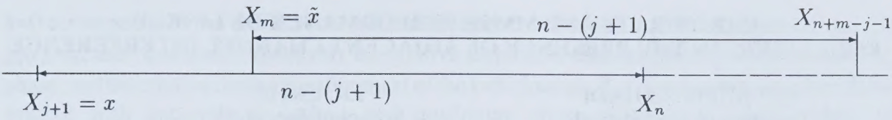


Figure 1: Schematic of Splitting Construction for $m_{j+1}(x) = m$ (assuming $m < n$.)

Acknowledgments

Michael Fu was supported in part by a General Research Board Award from the University of Maryland at College Park. **Jian-Qiang Hu** was supported in part by the National Science Foundation (NSF) under grants EID-9212122 and DDM-9212368. **Leyuan Shi** was supported in part by the Office of Naval Research (ONR) and the Defense Advanced Research Projects Agency (DARPA) under contract N00014-91-C-0195 to Honeywell and the Computer Science Department of the University of Maryland at College Park.

References

[1] Fox, B.L. and Glasserman, P., "Estimating Derivative Via Poisson's Equation," *Probability in the Engineering and Information Sciences,* **5**, No. 4, pp. 415-428, 1991.

[2] Fu, M.C. and Hu, J.Q., "Extensions and Generalizations of Smoothed Perturbation Analysis in a Generalized Semi-Markov Process Framework," to appear in *IEEE Trans. Automatic Control*, 1992.

[3] Fu, M.C. and Hu, J.Q., "Smoothed Perturbation Analysis Derivative Estimation for Markov Chains," submitted to *Operations Research Letters*, 1991.

[4] Fu, M.C. and Hu, J.Q., "Smoothed Perturbation Analysis for Queues with Finite Buffer," to appear in *Queueing Systems: Theory and Applications*, 1993.

[5] Glynn, P.W., "Likelihood Ratio Gradient Estimation: An Overview," *Proceedings of the Winter Simulation Conference,* pp. 366-374, 1987.

[6] L'Ecuyer, P., "A Unified View of the IPA, SF and LR Gradient Estimation Techniques," *Management Science,* **36**, No. 11, pp. 1364-1383, 1990.

[7] Reiman, M.I. and Weiss, A., "Sensitivity Analysis for Simulations via Likelihood Ratios," *Operations Research,* **37**, No. 5, pp. 830-844, 1989.

[8] Rubinstein, R.Y., "Sensitivity Analysis and Performance Extrapolation for Computer Simulation Models," *Operations Research,* **37**, No. 1, pp. 72-81, 1989.

[9] Shi, L.Y., "Discontinuous Perturbation Analysis of Discrete Event Dynamic Systems," submitted to *IEEE Transactions on Automatic Control*, 1992.

[10] Zhang, B. and Ho, Y.C., "Performance Gradient Estimation for Very Large Finite Markov Chains," *IEEE Transactions on Automatic Control*, **36**, No. 11, pp. 1218-1227, 1991.

ERROR RATE AND MMSE PERFORMANCE OF LINEAR EQUALIZERS IN THE PRESENCE OF ADJACENT CHANNEL INTERFERENCE

M. TAHERNEZHADI
Department of Electrical
Engineering
Northern Illinois University
DeKalb, IL 60115

J.R. CRUZ
School of Electrical
Engineering
The University of Oklahoma
Norman, OK 73019

ABSTRACT

This paper develops an analytical technique to evaluate the error rate and the *minimum-mean-square-error* (MMSE) performance of linearly equalized M-ary quadrature amplitude (QAM) and quadrature partial response signaling (QPRS) in the presence of adjacent channel interference. Both symbol and fractionally spaced linear equalizers (SSE and FSE) are considered. The tap gains of these equalizers are evaluated in the MMSE sense. The transfer function and MMSE of the infinite tap SSE are derived in the presence of interfering channels. The underlying structure of the infinite-tap SSE is used to develop an infinitely long FSE. Numerical results indicate that the proposed structure is not yielding the same MMSE as obtained by a sufficiently long FSE. A frequency domain analysis in conjunction with singular value decomposition (SVD), reveals that the case for the failure of the proposed structure is due to a built-in rank reduced property.

I. Introduction

High level quadrature amplitude modulations are acutely sensitive to distortions engendered by channel imperfections, additive noise and ACI. The mathematical formulation of adaptive equalizers is largely limited to investigate dispersive channels with stationary Gaussian noise [5]-[7]. The problem of how to handle degradations introduced by ACI in the analysis has not been addressed. Although considerable work has been done on the design of receiver filters in an ACI environment [8]-[11], they have not approached the formulation from the adaptive equalization standpoint- with the exception of the work in [7], where it was assumed that the symbol estimates of interfering channels are known to the receiver. In here, however, no knowledge of estimated symbols of interfering channels is assumed.

The objective of this paper is to provide a mathematical formulation to analyze the performance of M-ary QAM and QPRS digital communication systems in the presence of ACI by employing SSE and FSE linear equalizers. The MMSE criterion is used for the determination of the tap gains of a fixed equalizer. Satisfactory performance of an equalizer in the presence of an interfering channel means that more bits/sec can be transmitted for a given bandwidth, i.e., it leads to higher transmission capacity.

The error rate computation technique due to Helstrom [1], which was originally devised for binary symbols in an ISI environment, is extended to M-ary QAM and QPRS systems undergoing the deleterious contributions of ACI, and additive white Gaussian noise. This technique provides an expeditious way to evaluate the error probability in the presence of a large number of random variables; see [1] and [2] for further discussions.

This paper is organized as follows. In Section II, the transfer function and the MMSE of the infinite tap SSE and FSE are derived in the presence of interfering channels. Section III presents numerical results and discussions on the performance of SSE and FSE.

II. Analytical Formulation

The receiver structure consisting of a front-end receiver filter, a sampler, and a finite-tap SSE/FSE is shown in Fig. 1. The normalized baseband received signal for M-ary QAM and QPRS system prior to the equalizer and after the front-end receiver filter is given by [2], [3]

$$r(t)=\sum_{k}\rho_k\sum_{i}\beta_{ki}g_k(t-\tau_k-iT_k)e^{j\alpha_{ki}}+n_1(t), \qquad (1a)$$

where

$$\alpha_{ki}=2\pi f_{sk}(\tau_k+iT_k)+\alpha_k, \qquad (1b)$$

k=0 corresponds to the desired channel and otherwise to the k-th interfering channel; ρ_k, f_{sk}, $g_k(t)$, α_k, and τ_k denote respectively the relative amplitude, channel spacing, system response, phase, and the relative timing misalignment of the k-th channel; β_{ki} is the quadrature information symbol with uncorrelated inphase and quadrature phase components taking values of $\pm 1, \pm 3, \ldots, \pm(\sqrt{M}-1)$ with equal probabilities and variance σ_a^2; T_k is the symbol duration of the k-th channel with the assumption that $T_k = T_o = T = 1$; and n(t) is the additive Gaussian noise at the output of the front-end receiver filter. The transmitter and receiver filters are assumed to be square-root of Nyquist and duobinary filters, respectively for M-ary QAM and QPRS systems. In addition, for QPRS systems, the precoding technique presented in [12] is employed in the transmitter in order to circumvent the knowledge of the previous symbols in the receiver.

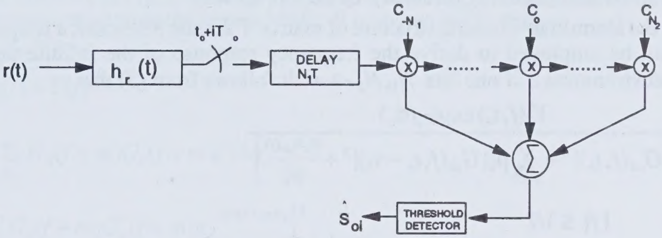


Fig. 1. Finite tap SSE/FSE receiver structure.

Considering an SSE/FSE with $N_1 + N_2 + 1$ number of taps, an estimate of the m-th transmitted symbol is obtained by

$$\hat{S}_{om} = \sum_{n=-N_1}^{N_2} c_n r(t_o + m - nT') . \tag{2}$$

where t_o is the sampling epoch, $T' = 1$, and K/M_o (K and M_o are relative prime with respect to each other) for SSE and FSE, respectively. Moreover, $S_{om} = \beta_{om}$ and $S_{om} = \beta_{om} + \beta_{o,m-1}$ denote the complex-valued desired symbol, respectively for *M*-ary QAM and QPRS systems. The orthogonality principle is employed to evaluate the tap gains c_n. The MMSE tap gains are given by

$$C = \Gamma^{-1} f \tag{3}$$

where

$$\Gamma_{(i,j)} = \frac{R(i,j)}{2\sigma_a^2} , \tag{4}$$

$$(f_i)_{QAM} = g_o^*(t_o - iT') e^{-j\alpha_o} . \tag{5a}$$

$$(f_i)_{QPRS} = (g_o^*(t_o - i) + g^*(t_o - i + 1)) e^{(-j\alpha_o)} , \tag{5b}$$

and

$$R(i,j) = R_o(t_o,i,j) + \sum_{k \neq 0} \rho_k^2 R_k(t_o,i,j,\tau_k) + R_n(i-j) , \tag{6}$$

$$R_n(i-j) = N_o \int_{-\infty}^{\infty} |H_R(f)|^2 \exp(j2\pi f(i-j)T') df , \tag{7a}$$

$$R_o(t_o,i,j) = 2\sigma_a^2 \sum_l g_o(t_o + l - iT') g_o^*(t_o + l - jT') , \tag{7b}$$

$$R_k(t_o,i,j,\tau_k) = 2\sigma_a^2 \sum_l g_k(t_o+l-\tau_k-iT')g_k^*(t_o+l-\tau_k-jT'). \tag{7c}$$

and the MMSE is given by [2]

$$\begin{aligned} |\varepsilon|^2 &= E(\varepsilon_m^*(S_{om}-\hat{S}_{om})) \\ &= E(\varepsilon_m^* S_{om}) \\ &= 2\sigma_a^2(v_o - f^H c), \end{aligned} \tag{8}$$

where $v_o = 1$ and 2 respectively for M-ary QAM and QPRS.

Owing to the Hermitian-Toeplitz structure of matrix Γ for the SSE case, a frequency domain approach can be employed to derive the frequency response of the infinite-tap SSE in an interfering environment . If one lets $N_1, N_2 \rightarrow \infty$, it follows from (3) that

$$C(f) = \frac{Y_o^*(f,t_o)\exp(-j\alpha_o)}{\left[|G_{ao}(f,t_o)|^2 + \sum\limits_{k\neq 0} \rho_k^2 |G_{ak}(f,t_o-\tau_k)|^2 + \frac{N_o H_{aR}(f)}{2\sigma_a^2} \right]} \qquad |f| \leq 1/2 \tag{9}$$

where

$$H_{aR}(f) = \sum_i |H_R(f-i)|^2 \qquad |f| \leq 1/2 \tag{10}$$

and $Y(f,t_o) = G_{ao}(f,t_o)$ and $G_{ao}(f,t_o) + G_{ao}(f,t_o+1)$, respectively for M-ary QAM and QPRS. Note that $G_{ak}(f,t_o-\tau_k)$ denotes the aliased (or folded) spectrum of the k-th channel and is given by

$$G_{ak}(f,t_o-\tau_k) = \sum_i G_k(f-i)\exp(j2\pi(t_o-\tau_k)(f-i)). \tag{11}$$

The function $H_{aR}(f)$ is the aliased power spectrum of the front-end receiver filter. Its value is 1 for M-ary QAM; this is due to the well-known Nyquist I criterion [12, pp. 334-338].

The expression for the infinite-tap equalizer given in (9) can be decomposed into three components in cascade: (1) a filter with frequency response $Y_o^*(f)\exp(-j\alpha_o)$, i.e., a matched filter which is matched to the pulse $y_o(t)\exp(j\alpha_o)$; (2) a symbol rate sampler, which is required in order to produce the aliased spectrum given in the numerator of (9); (3) a filter with the frequency response $1/X(f)$, where $X(f)$ is the denominator of (9). Moreover, since $1/X(f)$ is a periodic function in the variable f, one can realize this filter by an infinite-tap transversal filter with symbol spaced taps. Thus, taking into account the front-end receiver filter, the overall receiver therefore consists of a matched filter, an infinite-tap symbol-spaced transversal equalizer, and a symbol-rate sampler. This is the well-known optimum receiver reported in the literature, e.g., [13]-[16]. It is worth noting that the above-mentioned optimum linear receiver structure has been derived [15], [16], albeit in the case of no interfering channels, in a completely different setting without assuming any a priori knowledge about the receiver structure other than being linear. The problem then consists of finding a linear receiver filter such that the output MSE is minimized. The aforementioned optimum linear filter in turn can be derived by employing standard variational techniques.

The MMSE of the infinite-tap equalizer follows from (8) and (9), and it can be expressed as

$$\varepsilon^2 = 2\sigma_a^2 \int_{-1/2}^{1/2} (v_o - C(f)Y_o(f,t_o))df$$

$$= 2\sigma_a^2 \int_{-1/2}^{1/2} df\,[v_o - \frac{|Y_{ao}^*(f,t_o)|^2}{\left[|G_{ao}(f,t_o)|^2 + \sum_{k\neq 0} \rho_k^2 |G_{ak}(f,t_o-\tau_k)|^2 + \frac{N_o H_{aR}(f)}{2\sigma_a^2}\right]}] \tag{12}$$

Inspired by the optimum infinite-tap SSE, a structure for the infinitely long FSE is proposed. This structure is shown in Fig. 2, where Z(f) is an infinitely long SSE that needs to be determined in such a way as to minimize the output MSE. It is shown in [2] that

$$Z(f) = \frac{P_{G_o}(f)}{P_{G_o}^2(f) + T(f)}, \tag{13}$$

where

$$T(f) = N_o \sum_m |H_R(f+m)G_o(f+m)|^2 / 2\sigma_a^2$$

$$+ \sum_{k\neq 0} \rho_k^2 \left| \sum_m G_k(f+m)G_o^*(f+m)e^{-j2\pi(f+m)\tau_k} \right|^2, \tag{14}$$

$$P_{G_o}(f) = \sum_m |G_o(f+m)|^2, \tag{15}$$

and N_o is the power of the zero mean Gaussian noise. The output MMSE is given by

$$\varepsilon_{QAM}^2 = 2\sigma_a^2 \int_{-1/2}^{1/2} \left[|1 - Z(f)P_{G_o}(f)|^2 + T(f)|Z(f)|^2\right] df. \tag{16}$$

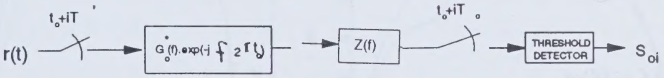


Fig. 2. Proposed infinite tap FSE structure.

Numerical results show that the proposed infinitely long FSE fails to yield the *matched filter bound* (MFB) as achieved by a sufficiently long finite-tap FSE in the presence of timing error. This shortcoming of the proposed structure is exacerbated in the presence of ACI, i.e., the achieved MMSE is significantly larger than the asymptotic MMSE obtained by a finite-tap FSE. For example, for a normalized channel spacing (f_s) of 0.90, the asymptotic MMSE is 1.1 dB as compared to 7.6 dB (evaluated numerically) for the proposed structure. In what follow, by employing a frequency domain analysis [2] in conjunction with SVD, it will be shown that the failure of the proposed structure is due to a built-in rank reduced property in system of equations for the infinitely long FSE's tap gains.

The set of equations in terms of the tap coefficients of an infinitely long 1/M-FSE, in the presence of noise and one interfering channel, can be expressed in the frequency domain as [2]

$$(G(f) + N(f))c(f) = g_o(f) \qquad |f| \le 0.5, \tag{17}$$

where

$$G(f) = g_o(f)g_o^H(f) + g_1(f)g_1^H(f), \tag{18a}$$

$$N_{ik}(f) = \sigma_n^2 \sum_m |H_R(f-m)|^2 \exp(j2\pi(f-m)(i-k)/M)$$

$$i,k = 1,2,\ldots,M-1, \tag{18b}$$

$$g_{ki}(f) = G_{ka}(f, i/M) \qquad k = 0, 1 \text{ and } i = 1, \ldots, M \tag{18c}$$

$$c_i(f) = C_a(f, i/M) \quad i = 1, \ldots, M , \tag{19}$$

C(f) is the transfer function of the equalizer. In the particular case of no noise and no interfering channels, it can be shown that matrix G has rank one. By employing SVD, one can arrive at the following decomposition for the $M \times M$ Hermitian matrix $(G + N)$ for any given frequency

$$G(f) + N(f) = \sum_{i=1}^{M} \lambda_i(f) u_i(f) u_i^H(f) , \tag{20}$$

where $\lambda_i(f)$ and $u_i(f)$ are, respectively, the i-th eigenvalue and eigenvector. Furthermore, owing to the decomposition of G in (18a), it is clear that the matrix G can be at most of rank two. At the particular frequency $f = 0$, however, it can be easily shown that, for M-ary QAM systems employing bandlimited Nyquist filters, the matrix $G + N$ is not of full rank regardless of the channel spacing parameter, f_s. In other words, at this frequency, (17) cannot be solved to uniquely determine the $M \times 1$ vector $c(f)$. However, with the help of SVD, one can approximate the $M \times M$ matrix G+N by its first eigenvalue and the corresponding eigenvector:

$$G + N \approx \lambda_1 u_1 u_1^H . \tag{21}$$

Now, by utilizing this approximation in (16), the unknown vector, c, can be obtained as

$$c \approx \frac{1}{\lambda_1} u_1 u_1^H g_o . \tag{22}$$

The objective now is to construct the transfer function of the infinitely long 1/M-FSE from its aliases, i.e, the aliases $C_a(f, i/M)$ (i=1,..., M) obtained from (22) must be employed to construct C(f). This is done as follows.

The tap gains c_n are related to the transfer function of the equalizer by

$$c_n = \int_{-M/2T}^{M/2T} C(f) \exp(j 2\pi f n/M)\, df$$

$$= \int_{-1/2}^{1/2} C_a(f, n/M) df , \tag{23}$$

where $C_a(f, n/M)$ is the aliased spectrum of C(f), and is given by

$$C_a(f, n/M) = \sum_m C(f + m) \exp(j 2\pi (f + m) n/M) . \tag{24}$$

Furthermore, by decomposing n as

$$n = q_n M + r_n \qquad r = 0, \ldots, M - 1 , \tag{25}$$

one has

$$C_a(f, n/M) = C_a(f, r_n/M) \exp(2 j \pi f q_n) \qquad r_n = 0, \ldots, M - 1 . \tag{26}$$

Thus, the tap coefficients can be computed from the aliased spectra as

$$c_n = \int_{-1/2}^{1/2} C_a(f, r_n/M) \exp(j 2\pi f q_n) df . \tag{27}$$

Having computed c_n, one can construct C(f) straightforwardly, i.e.,

$$C(f) = \sum_n c_n \exp(j 2\pi f n/M) . \tag{28}$$

As an example, two systems with rolling factor parameters of 0.25 and 1.0, with $f_s = 0.90$, are considered. The corresponding transfer functions, synthesized by a 1/4-FSE of length 21 (N=10), are shown in Fig. 3. The transfer functions, obtained from the proposed structure and eigenanalysis, are shown in Fig. 4. It is seen that the eignanalysis provides a close approximation to the proposed structure. That is to say that one can view the proposed structure as being constructed from the dominant eigenvalue and its corresponding eigenvector. In other words, the built-in eigenapproximation plagues the performance of the proposed structure, leading to a substantially inferior MMSE.

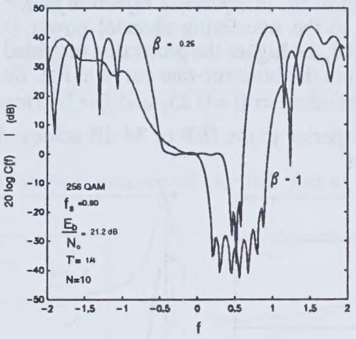
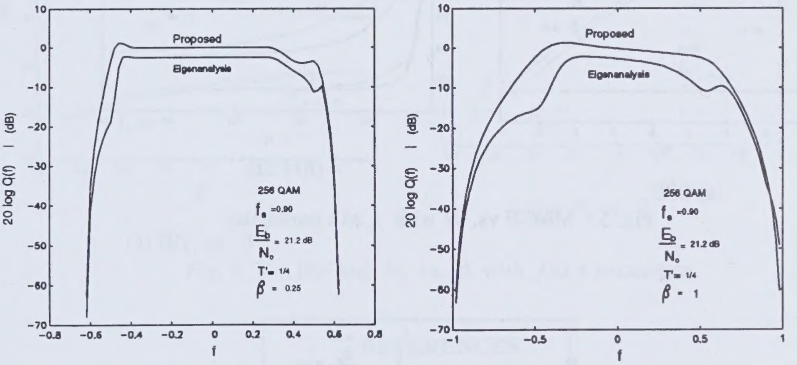


Fig. 3. 1/4-FSE transfer function with $N = 10$ and $f_s = 0.90$.



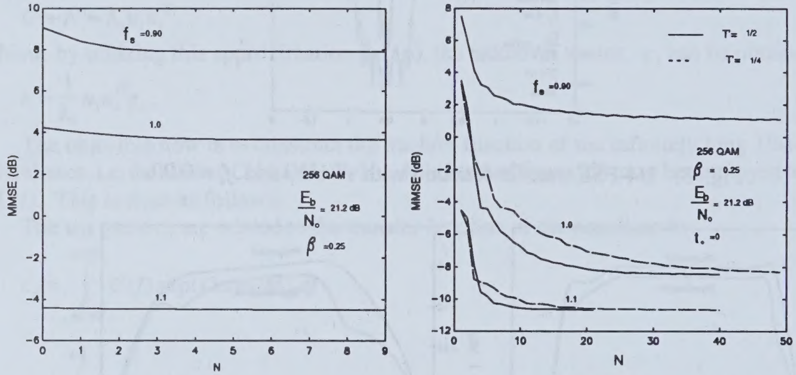
(a) $\beta = 0.25$ (b) $\beta = 1$

Fig. 4. Transfer function of the infinitely-long 1/4-FSE obtained from the proposed structure and the eigenanalysis.

III. Numerical Results and Discussions

It is seen in Fig. 5a that an SSE is quite incapable of reducing the MMSE in the presence of ACI. Although the equalizer exhibits a fast decay to its asymptotic performance, the reduction in the MMSE is relatively insignificant; for example, it is less than 1.2 dB for a normalized channel spacing of 0.9, when measured from MMSE at $N = 0$. The FSE, however, though exhibiting a slower decay, provides substantial reduction in the MMSE as seen in Fig. 5b Although the FSE offers adequate protection against ACI for the normalized channel spacings of 1.0 and 1.1 in this particular example, the asymptotic MMSE does not approach the MFB as the normalized channel spacing is reduced. This behavior is shown in Fig. 6, where a relatively large $N = 20$ is used to depict the almost asymptotic MMSE performance of a 1/2-FSE. Thus, the best linear equalizer becomes incapable of achieving the MFB as the channel spacing parameter is reduced. Nonetheless, the FSE performance in the presence of ACI is independent of the timing parameter, t_o as shown in Fig. 7. In short, considering a 256-ary QAM system with a rolling factor of 0.25, a sufficiently long FSE yields the match filter bound as long as $f_s \geq 1.1$. That is, the transmission efficiency can be increased from 6.4 bits/sec/Hz to 7.27 bits/sec/Hz (8/1.1) when employing a sufficiently long FSE. That is an increase of 9.47% in the transmission efficiency over the unequalized receiver [2].

In addition, Fig. 8a illustrates the *interference rejection factor* (IRF), defined as the ratio of the desired channel power to the interfering channel power [17], versus the rolling factor parameter. The higher the IRF, the higher the protection provided by the equalizer against ACI. For example, it is evident from the bit-error-rate curve in Fig. 8b that an IRF of 49 dB offered by a 41-tap 1/2-FSE (corresponding to $\beta = 0.25$, and $f_s = 1.1$) leads to an average bit-error-rate of $10^{-3.8}$. This IRF is far superior to the IRF of 34 dB achieved at $f_s = 1.0$, which yields an average bit-error-rate $10^{-2.75}$.



(a) SSE

(b) FSE

Fig. 5. MMSE vs. N with f_s as a parameter.

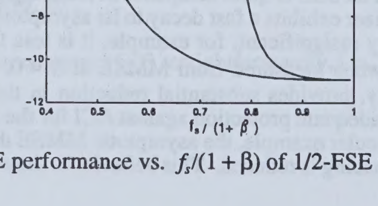


Fig. 6. MMSE performance vs. $f_s/(1+\beta)$ of 1/2-FSE with $N = 20$.

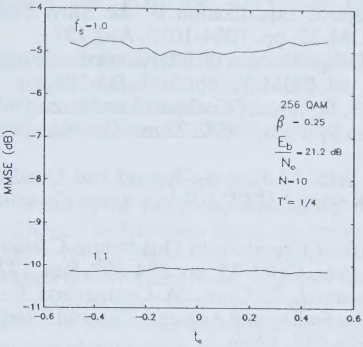
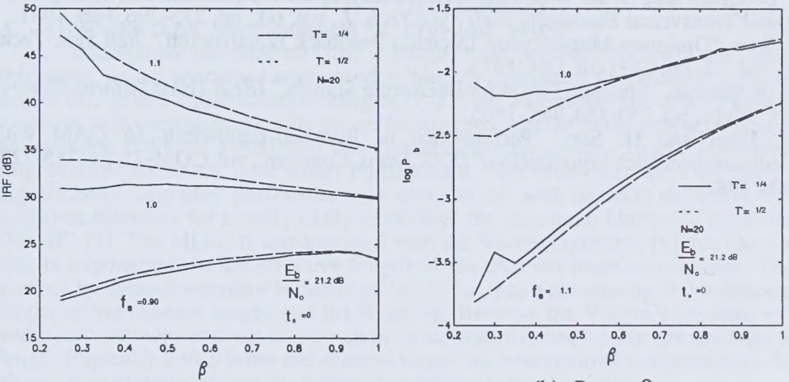


Fig. 7. MMSE performance vs. t_o with $f_s$as a parameter.



(b) P_b vs. β

(a) IRF vs. β

Fig. 8 The IRF and P_b vs. β with $f_s$as a parameter.

REFERENCES

[1] C. W. Helstrom, "Calculating Error Probabilities for Intersymbol and Cochannel Interference," *IEEE Trans. Commun.*, vol. COM-34, pp. 430-435, May 1986.

[2] M. Tahernezhadi, "Theory and Performance of Equalized and Unequalized QAM and QPRS Systems in the Presence of Frequency Selective Fading and Interfering Channels," Ph.D. Dissertation, The University of Oklahoma, May 1991.

[3] M. Tahernezhadi, and J. R. Cruz, "Frequency Selective Fading and Adjacent Channel Interference Performance of MSK and SFSK," *IEEE Trans. Commun.*, vol. COM-38, pp. 102-108, Jan. 1990.

[4] S. H. Qureshi, "Adaptive Equalization," *Proc. IEEE*, vol. 73, pp. 1349-1387, Sep. 1985.

[5] P. Monsen, "Feed Back Equalization for Fading Dipersive Channels," *IEEE Trans. Inform. Theory*, vol. IT-17, pp. 56-64, Jan. 1971.

[6] P. Monsen, "Adaptive Equalization of the Slow Fading Channels," *IEEE Trans. Commun.*, vol. COM-22, pp. 1064-1075, Aug 1974.

[7] ______, "MMSE Equalization of Interference on Fading Diversity Channels," *IEEE Trans. Commun.*, vol. COM-32, pp. 5-12, Jan 1984.

[8] J. R. Cruz and R. S. Simpson, "Cochannel and Intersymbol Interference in Quadrature Carrier Modulation Systems," *IEEE Trans. Commun.*, vol. COM-29, pp. 285-297, Mar. 1981.

[9] I. Korn and B. Seth, "Adjacent-Channel and Quadrature Channel Interference in Minimum Shift Keying," *IEEE J. Select. Areas Commun.*, vol. SAC-1, pp. 21-28, Jan. 1982.

[10] I. Kalet, "A Look at Crosstalk in Quadrature-Carrier Modulation Systems," *IEEE Trans. Commun.*, vol. COM-25, pp. 884-892, Sep. 1977.

[11] M. Tahernezhadi and J. R. Cruz, "A Comparison Between SFSK and MSK With Frequency Selective Fading and Adjacent Channel Interference," in *Proc. 24th Midwest Symposium*, pp. 624-628, 1986.

[12] J. G. Proakis, *Digital Communications*, New York: McGraw-Hill Inc., 1983.

[13] M. S. Muller and J. Salz, "A Unified Theory of Data-aided Equalization," *Bell Syst. Tech. J.*, vol. 6, pp. 2023-2038, Nov. 1981.

[14] R. D. Gitlin and S. B. Weistein, "Fractionally-Spaced Equalization: An Improved Digital Transversal Equalizer," *Bell Syst. Tech. J.*, vol. 60., pp. 275-296, Feb. 1981.

[15] J. Salz, "Optimum Mean-Square Decision Feedback Equalization," *Bell Syst. Tech. J.*, vol. 52, pp. 1431-73, Oct. 1973.

[16] D. A. George, "Matched Filters for Interfering Signals," *IEEE Trans. Inform. Theory*, vol. IT-11, pp. 153-154, Jan. 1965.

[17] A. Fihel and H. Sari, "Performance of Reduced-Bandwidth 16 QAM with Decision-Feedback Equalization," *IEEE Trans. Commun.*, vol. COM-35, pp. 715-723, July 1987.

Bayesian Techniques for Known Channel Deconvolution*

Gen-kwo Lee, Saul B. Gelfand and Michael P. Fitz
School of Electrical Engineering
Purdue University
West Lafayette, IN 47907-1285

Abstract

This paper introduces a family of fixed lag smoothing filters for known channel deconvolution called Bayesian filters. The Bayesian filters are derived using a new approach to recursive minimum mean square error estimation for nonGaussian processes. The family of Bayesian filters interpolate between the optimum fixed lag linear filter (i.e., the Kalman filter) and the optimum fixed lag symbol-by-symbol detector in both performance and complexity. The Bayesian filter distinguishes itself from other algorithms for known channel deconvolution by decoupling the exponential complexity from both the channel length and the filter lag. Simulation results are given which characterize the performance of the Bayesian filter in channels producing severe ISI and with large constellations (i.e., 16-QAM).

I. Introduction

A. Overview and Shortcomings of Previous Work

Intersymbol interference (ISI) is a significant problem in digital communications. In this paper, we are concerned with uncoded, linearly modulated transmission systems subject to severe ISI. Linear equalization techniques [7, 11] are widespread due to their simplicity, but in channels with deep spectral nulls the performance can be far from optimal. Decision feedback equalization (DFE) is a nonlinear equalization technique with a simple implementation and improved performance over linear equalization. Unfortunately the DFE can still have a significantly degraded performance in comparison with optimal detection. The optimal sequence estimator for equally likely symbols is the maximum likelihood sequence estimator (MLSE) [5]. The MLSE is implemented with the Viterbi algorithm [6] and has a complexity that is exponential in the effective length of the channel impulse response. The optimum symbol-by-symbol estimator is described in [1], but has a complexity that is exponential in the larger of the channel length and the filter lag. Because the Viterbi algorithm only requires additional memory and not computation to support a decoding lag greater than the channel length (typically 2 to 5 times the channel length for near optimal performance), the MLSE is almost always prefered over the optimal symbol-by-symbol estimator.

The complexity of the MLSE has led to a considerable amount of research seeking to achieve near optimal performance at reduced complexity [3, 4, 9]. One approach is to use a prefilter to produce an equivalent channel having a shorter impulse response [9]. Two recently proposed sequence estimators, namely reduced-state sequence estimation (RSSE) [4] and delayed decision-feedback sequence estimation (DDFSE) [3], provide a tradeoff between performance and complexity. The major problem with these algorithms is that no principled design technique exists for selecting and adapting the reduced state partition on a general channel.

Complexity reduction for the optimal symbol-by-symbol estimator has been attempted by employing limited decision feedback in both a recursive [1] and nonrecursive [12] fashion. The resulting algorithms have a complexity which is still exponential in the filter lag, and hence may not always provide the desired performance at a reasonable complexity.

This paper proposes a new family of deconvolution algorithms called Bayesian filters, which bridge the gap between linear equalization algorithms and the optimum symbol-by-symbol detection algorithm. The exponential complexity of these algorithms is decoupled from both the channel length and filter lag. Also these algorithms work well with large constellations (e.g., 16-QAM) with significantly less computational complexity than the MLSE.

The paper is organized as follows. Section II presents the family of Bayesian filter algorithms. Section III presents a simulation based performance assessment of these algorithms and a comparison to other common algorithms (i.e., DFE, MLSE, linear

---

* Research partially supported by National Science Foundation under Grants NCR-9010239 and NCR-9115820.

equalization). Section IV concludes.

B. Analytical Models for Deconvolution

Fig. 1 illustrates a baseband equivalent model of digital communication over a time-dispersive channel. In this model complex-valued M-ary signals, d_k, pass through a linear time-varying FIR channel whose equivalent discrete-time impulse response is denoted by $c_i(k)$, $0 \le i \le L-1$. The filtered signal is corrupted by independent complex additive white Gaussian noise (AWGN), v_k, with variance N_0. Hence, the channel output, sampled at symbol rate, is

$$y_k = \sum_{i=0}^{L-1} c_i(k) d_{k-i} + v_k, \quad k = 1, 2, \ldots \tag{1}$$

We assume the d_k are i.i.d. random variables with mass $p_d(\cdot)$, mean m_d and variance σ_d^2, and are taken from a constellation Ω of size M.

This paper derives high performance estimators of the transmitted data sequence based on the model in (1) where the channel impulse response is known. Let $Y(k)$ denote the set of observations up to time k, i.e., $Y(k) = \{y_1, y_2, \ldots, y_k\}$. More precisely, this work develops K-lag smoothed estimators of d_{k-K}

$$\hat{d}_{k-K} = \varphi_k(Y(k)). \tag{2}$$

If the estimate $\hat{d}_{k-K}$ is not in Ω, then a minimum (Euclidean) distance rule is used to obtain an estimate $\tilde{d}_{k-K}$ in Ω (see Fig. 1).

II. The Bayesian Filter

The optimal K-lag filter, i.e., the K-lag maximum aposteriori probability (MAP) symbol-by-symbol detector [1], is based on finding an expression for the mass function $p(d_{k-K}|Y(k))$. Here and in the sequel we will only consider the important case where $K+1 \ge L$, i.e., the filter lag is no less than the channel length. Denoting $\mathbf{d}(k) = \{d_k, \ldots, d_{k-K}\}$, the mass function $p(\mathbf{d}(k) \mid Y(k-1))$ is recursively computed, and $p(d_{k-K}|Y(k))$ is then obtained from $p(\mathbf{d}(k) \mid Y(k-1))$ and y_k. The recursion for $p(\mathbf{d}(k) \mid Y(k-1))$ is a straightforward application of Bayes rule. The (sufficient) statistic $p(\mathbf{d}(k) \mid Y(k-1))$ has dimension M^K.

The major drawback of the optimal K-lag filter is the exponential complexity in recursively computing the statistic $p(\mathbf{d}(k) \mid Y(k-1))$. Here we develop a family of reduced complexity suboptimal K-lag filters for known channel deconvolution: the Bayesian filters (BFs). The idea is to first partition the symbol vector $\mathbf{d}(k)$ into initial and final segments, and then approximate the final segment as a jointly Gaussian vector conditioned on the past data $Y(k-1)$ and the initial segment. These filters interpolate between the optimal K-lag linear filter (i.e., the Kalman filter (KF) [2]) and the optimal K-lag filter in both performance and complexity.

The development of the BF employs the MMSE criterion and seeks the conditional mean, i.e.,

$$\hat{d}_{k-K} = E\{d_{k-K}|Y(k)\}. \tag{3}$$

We remark that it is also possible in our framework to develop BFs which employ the MAP criterion and seek the conditional mode, at the cost of increased algorithm complexity and marginal performance gain; this variation on BFs and some simulation results are discussed in [8].

A. Algorithm Overview

The BF is developed from a state-space parameterization of the observation model in (1) given as

$$\begin{aligned}\mathbf{x}(k+1) &= \mathbf{A}\,\mathbf{x}(k) + \mathbf{w}(k)\\ y_k &= \mathbf{H}(k)\mathbf{x}(k) + v_k, \qquad k = 1, 2, \ldots\end{aligned} \tag{4}$$

where $\mathbf{A}$ is a shift matrix and

$$\mathbf{x}(k) = \left[x_0(k) \cdots x_K(k)\right]^T = \left[d_k \cdots d_{k-K}\right]^T, \tag{5a}$$

$$\mathbf{w}(k) = \left[w_0(k) \cdots w_K(k)\right]^T = \left[d_{k+1}, 0 \cdots 0\right]^T, \tag{5b}$$

$$\mathbf{H}(k) = \left[h_0(k) \cdots h_K(k)\right] = \left[c_0(k) \cdots c_{L-1}(k), 0 \cdots 0\right] \qquad (K \geq L-1). \tag{5c}$$

Let

$$\mathbf{x}^+(k) = \left[x_1(k) \cdots x_{J-1}(k)\right]^T = \left[d_{k-1} \cdots d_{k-J+1}\right]^T, \tag{6a}$$

$$\mathbf{x}^-(k) = \left[x_J(k) \cdots x_K(k)\right]^T = \left[d_{k-J} \cdots d_{k-K}\right]^T, \tag{6b}$$

$$\mathbf{H}^+(k) = \left[h_1(k) \cdots h_{J-1}(k)\right] = \left[c_1(k) \cdots c_{J-1}(k)\right], \tag{6c}$$

$$\mathbf{H}^-(k) = \left[h_J(k) \cdots h_K(k)\right] = \left[c_J(k) \cdots c_{L-1}(k), 0 \cdots 0\right] \qquad (K \geq J-1). \tag{6d}$$

Here $\mathbf{x}^+(k)$ and $\mathbf{x}^-(k)$ are the initial and final segments of the symbol vector referred to above (excluding the initial component $x_0(k) = d_k$). Hence we can write the observation y_k as

$$y_k = c_0(k)x_0(k) + \mathbf{H}^+(k)\mathbf{x}^+(k) + \mathbf{H}^-(k)\mathbf{x}^-(k) + v_k. \tag{7}$$

The three parameters J, K, L appear repeatedly in the sequel. Recall that L is the equivalent channel length and K is the filter lag length (and we only consider $K \geq L-1$). The parameter J ($1 \leq J \leq K+1$), which we shall call the chip length, will index the complexity and performance of the BFs. Typically we choose $J < \min\{K, L\}$.

We also require the following definitions for various conditional mean vectors and covariance matrices

$$\hat{\mathbf{x}}^\bullet(k|l;\mathbf{z}) = E\{\mathbf{x}^\bullet(k)|Y(l),\mathbf{z}\}, \qquad \mathbf{P}^\bullet(k|l;\mathbf{z}) = Cov\{\mathbf{x}^\bullet(k)|Y(l),\mathbf{z}\}, \tag{8}$$

where $\mathbf{z}$ is either a random or null vector, and $\bullet$ is either +,- or null. We shall use subscripts to denote the components of the conditional mean vectors and covariance matrices, e.g.,

$$\hat{x}_i(k|l;\mathbf{z}) = E\{x_i(k)|Y(l),\mathbf{z}\}, \qquad \mathrm{p}_{i,j}(k|l;\mathbf{z}) = Cov\{x_i(k), x_j(k)|Y(l),\mathbf{z}\}. \tag{9}$$

In this notation the desired estimate is

$$\hat{d}_{k-K} = \hat{x}_K(k\,|\,k) = E\{x_K(k)|Y(k)\}. \tag{10}$$

The estimate in (10) requires solving a nonGaussian MMSE filtering problem, as the process noise $\mathbf{w}(k)$ is nonGaussian (since d_k is discrete-valued). Our approach is somewhat related to the 1-step optimal filter for nonGaussian measurement noise described in [10], in that we approximate a conditional state distribution at each time step in such a way as to yield a recursive (but in general suboptimal) filter structure. We first examine the KF and then derive the BF in this setting.

The KF can be viewed as an approximate solution to our nonGaussian MMSE filtering problem. From this perspective, the KF is derived by assuming that the following approximation is valid at each time step k:

(A0): $\mathbf{x}(k)$ is conditionally Gaussian given $Y(k-1)$.

Under (A0), $\hat{\mathbf{x}}(k\,|\,k)$ and $\mathbf{P}(k\,|\,k)$ can be computed in terms of $\hat{\mathbf{x}}(k\,|\,k-1)$, $\mathbf{P}(k\,|\,k-1)$ and y_k. $\hat{\mathbf{x}}(k+1\,|\,k)$ and $\mathbf{P}(k+1\,|\,k)$ can then be computed from $\hat{\mathbf{x}}(k\,|\,k)$ and $\mathbf{P}(k\,|\,k)$ using the state space model (4), giving the familiar KF recursion. In the KF, $\hat{\mathbf{x}}(k\,|\,k)$ is a linear function of y_k, and $\mathbf{P}(k\,|\,k)$ does not depend on y_k. Although the KF can be used for equalization of digital communications channels [7], it (like other linear equalizers) does not always perform well.

The BF proposed here is derived by relaxing the KF approximation. Instead of (A0), we assume the following (weaker) approximation is valid at each time step k:

(A1): $\mathbf{x}^-(k)$ is conditionally Gaussian given $\left(Y(k-1), \mathbf{x}^+(k)\right)$.

This approximation depends on the chip length J $(1 \le J \le K+1)$, i.e., on the length of the initial segment $(x_0(k), \mathbf{x}^+(k))$ of the partitioned symbol vector. Note that we get successively weaker approximations as J increases from 1 to K+1. The KF can be thought of as the J=0 BF and the optimal K-lag filter (with MMSE criterion) is the J=K+1 BF.

In the sequel we shall make the (strictly invalid) assumption that the approximation (A1) is true. For the $J = 1$ case, $\hat{\mathbf{x}}(k \mid k)$ and $\mathbf{P}(k \mid k)$ can be computed in terms of $\hat{\mathbf{x}}(k \mid k-1)$, $\mathbf{P}(k \mid k-1)$ and y_k. $\hat{\mathbf{x}}(k+1 \mid k)$ and $\mathbf{P}(k+1 \mid k)$ can then be computed from $\hat{\mathbf{x}}(k \mid k)$ and $\mathbf{P}(k \mid k)$ using the model in (4), giving the $J = 1$ BF recursion. We note, however, that unlike the KF, in the $J = 1$ BF $\hat{\mathbf{x}}(k \mid k)$ is a nonlinear function of y_k, and $\mathbf{P}(k \mid k)$ depends on y_k (also nonlinearly). Fig. 2(a) shows the flow diagram for the BF with J=1. For the $1 < J \le K$+1 case, $\hat{\mathbf{x}}(k \mid k; \mathbf{x}^+(k+1))$, $\mathbf{P}(k \mid k; \mathbf{x}^+(k+1))$ and $p(\mathbf{x}^+(k+1) \mid Y(k))$ can be computed in terms of $\hat{\mathbf{x}}(k \mid k-1; \mathbf{x}^+(k))$, $\mathbf{P}(k \mid k-1; \mathbf{x}^+(k))$, $p(\mathbf{x}^+(k) \mid Y(k-1))$, and y_k. $\hat{\mathbf{x}}(k+1 \mid k; \mathbf{x}^+(k+1))$ and $\mathbf{P}(k+1 \mid k; \mathbf{x}^+(k+1))$ can then be computed from $\hat{\mathbf{x}}(k \mid k; \mathbf{x}^+(k+1))$ and $\mathbf{P}(k \mid k; \mathbf{x}^+(k+1))$ using the model in (4), giving the general BF recursion. Fig. 2(b) shows the flow diagram of the BF for $J > 1$.

B. Derivation of the Bayesian Filter

The derivation of the BF, like the KF, consists of a measurement and a prediction update.

Measurement Update for the Bayesian Filter

In this step we obtain the filtered estimate $\hat{\mathbf{x}}(k \mid k; \mathbf{x}^+(k+1))$, error covariance $\mathbf{P}(k \mid k; \mathbf{x}^+(k+1))$ and mass function $p(\mathbf{x}^+(k+1) \mid Y(k))$ in terms of the predicted estimate $\hat{\mathbf{x}}(k \mid k-1; \mathbf{x}^+(k))$, error covariance $\mathbf{P}(k \mid k-1; \mathbf{x}^+(k))$ and mass function $p(\mathbf{x}^+(k) \mid Y(k-1))$, and the current observation y_k.

First, we compute $\hat{\mathbf{x}}^-(k \mid k; x_0(k), \mathbf{x}^+(k))$ and $\mathbf{P}^-(k \mid k; x_0(k), \mathbf{x}^+(k))$ using standard properties of (conditionally) Gaussian random variables. In view of assumption (A1), $\left(\mathbf{x}^-(k), y_k\right)$ are jointly Gaussian conditioned on $\left(Y(k-1), x_0(k), \mathbf{x}^+(k)\right)$. Furthermore, from (7), the joint 2nd order moments of $\left(\mathbf{x}^-(k), y_k\right)$ conditioned on $\left(Y(k-1), x_0(k), \mathbf{x}^+(k)\right)$ are

$$E\left\{y_k \middle| Y(k-1), x_0(k), \mathbf{x}^+(k)\right\} = c_0(k)x_0(k) + \mathbf{H}^+(k)\mathbf{x}^+(k) + \mathbf{H}^-(k)\hat{\mathbf{x}}^-\left(k\middle|k-1; \mathbf{x}^+(k)\right) \tag{11a}$$

$$Var\left\{y_k \middle| Y(k-1), x_0(k), \mathbf{x}^+(k)\right\} = \mathbf{H}^-(k)\mathbf{P}^-\left(k\middle|k-1; \mathbf{x}^+(k)\right)\mathbf{H}^{-^H}(k) + \mathrm{N}_0 \tag{11b}$$

$$Cov\left\{\mathbf{x}^-(k), y_k \middle| Y(k-1), x_0(k)\, \mathbf{x}^+(k)\right\} = \mathbf{P}^-\left(k\middle|k-1; \mathbf{x}^+(k)\right)\mathbf{H}^{-^H}(k). \tag{11c}$$

Recall that if $\mathbf{u}$ and $\mathbf{z}$ are jointly Gaussian, then $E\{\mathbf{u}|\mathbf{z}\} = E\{\mathbf{u}\} + Cov\{\mathbf{u}, \mathbf{z}\}(Cov\{\mathbf{z}\})^{-1}(\mathbf{z} - E\{\mathbf{z}\})$ and $Cov\{\mathbf{u}|\mathbf{z}\} = Cov\{\mathbf{u}\} - Cov\{\mathbf{u}, \mathbf{z}\}(Cov\{\mathbf{z}\})^{-1}Cov\{\mathbf{u}, \mathbf{z}\}$. Setting $\mathbf{u} = \mathbf{x}^-(k), \mathbf{z} = y_k$ and using (11) gives

$$\begin{aligned}\hat{\mathbf{x}}^-\left(k\middle|k; x_0(k), \mathbf{x}^+(k)\right) &= \hat{\mathbf{x}}^-\left(k\middle|k-1; \mathbf{x}^+(k)\right) \\ &\quad + \mathbf{g}^-\left(k; \mathbf{x}^+(k)\right)\left(y_k - c_0(k)x_0(k) - \mathbf{H}^+(k)\mathbf{x}^+(k) - \mathbf{H}^-(k)\hat{\mathbf{x}}^-\left(k\middle|k-1; \mathbf{x}^+(k)\right)\right)\end{aligned} \tag{12a}$$

$$\mathbf{P}^-\left(k\middle|k; x_0(k), \mathbf{x}^+(k)\right) = \mathbf{P}^-\left(k\middle|k-1; \mathbf{x}^+(k)\right) - \mathbf{g}^-\left(k; \mathbf{x}^+(k)\right)\mathbf{g}^{-^H}\left(k; \mathbf{x}^+(k)\right)s^-\left(k; \mathbf{x}^+(k)\right) \tag{12b}$$

where the conditional filter gain $\mathbf{g}^-\left(k; \mathbf{x}^+(k)\right)$ and innovation variance $s^-\left(k; \mathbf{x}^+(k)\right)$ are given by

$$\mathbf{g}^-\left(k; \mathbf{x}^+(k)\right) = \mathbf{P}^-\left(k\middle|k-1; \mathbf{x}^+(k)\right)\mathbf{H}^{-^H}(k)s^{-^{-1}}\left(k; \mathbf{x}^+(k)\right) \tag{13}$$

$$s^-\left(k;\mathbf{x}^+(k)\right)=\mathbf{H}^-(k)\mathbf{P}^-\left(k|k-1;\mathbf{x}^+(k)\right)\mathbf{H}^{-H}(k)+\mathrm{N}_0. \tag{14}$$

Next, we compute $p(x_{J-1}(k)\,|\,Y(k),\mathbf{x}^+(k+1))$ and $p(\mathbf{x}^+(k+1)\,|\,Y(k))$. These posterior mass functions are given by Bayes rule as

$$p(x_{J-1}(k)\,|\,Y(k),\mathbf{x}^+(k+1))=\frac{f\left(y_k|Y(k-1),x_0(k),\mathbf{x}^+(k)\right)p\left(\mathbf{x}^+(k)|Y(k-1)\right)p(x_0(k))}{\sum\limits_{x_{J-1}(k)}f\left(y_k|Y(k-1),x_0(k),\mathbf{x}^+(k)\right)p\left(\mathbf{x}^+(k)|Y(k-1)\right)p(x_0(k))} \tag{15}$$

$$p(\mathbf{x}^+(k+1)\,|\,Y(k))=\frac{\sum\limits_{x_{J-1}(k)}f(y_k\,|\,Y(k-1),x_0(k),\mathbf{x}^+(k))p(\mathbf{x}^+(k)\,|\,Y(k-1))p(x_0(k))}{\sum\limits_{x_0(k),\mathbf{x}^+(k)}f(y_k\,|\,Y(k-1),x_0(k),\mathbf{x}^+(k))p(\mathbf{x}^+(k)\,|\,Y(k-1))p(x_0(k))} \tag{16}$$

Note that y_k conditioned on $\left(Y(k-1),x_0(k),\mathbf{x}^+(k)\right)$ is Gaussian with conditional mean and variance given by (11a) and (11b), respectively, and $p(\mathbf{x}^+(k)\,|\,Y(k-1))$ and $p(x_0(k))=p_d(x_0(k))$ are known.

Finally, we have

$$\hat{\mathbf{x}}(k\,|\,k;\mathbf{x}^+(k+1))=\left[\mathbf{x}^{+T}(k+1)\quad \hat{x}_{J-1}(k\,|\,k;\mathbf{x}^+(k+1))\quad \hat{\mathbf{x}}^{-T}(k\,|\,k;\mathbf{x}^+(k+1))\right]^T \tag{17a}$$

$$\mathbf{P}(k\,|\,k;\mathbf{x}^+(k+1))=\begin{bmatrix}\mathbf{0} & 0 & \mathbf{0}\\ \mathbf{o}^T & p_{J-1,J-1}(k\,|\,k;\mathbf{x}^+(k+1)) & \mathbf{p}_{J-1}^H(k\,|\,k;\mathbf{x}^+(k+1))\\ \mathbf{0} & \mathbf{p}_{J-1}(k\,|\,k;\mathbf{x}^+(k+1)) & \mathbf{P}^-(k\,|\,k;\mathbf{x}^+(k+1))\end{bmatrix} \tag{17b}$$

where, by standard results for iterated expection and covariance,

$$\hat{x}_{J-1}(k\,|\,k;\mathbf{x}^+(k+1))=E\left\{x_{J-1}(k)\,|\,Y(k),\mathbf{x}^+(k+1)\right\} \tag{18a}$$

$$\begin{aligned}\hat{\mathbf{x}}^-(k\,|\,k;\mathbf{x}^+(k+1))&=E\left\{\mathbf{x}^-(k)\,|\,Y(k),\mathbf{x}^+(k+1)\right\}\\ &=E\left\{\hat{\mathbf{x}}^-(k\,|\,k;x_0(k),\mathbf{x}^+(k))\,|\,Y(k),\mathbf{x}^+(k+1)\right\}\end{aligned} \tag{18b}$$

$$p_{J-1,J-1}(k\,|\,k;\mathbf{x}^+(k+1))=Var\left\{x_{J-1}(k)\,|\,Y(k),\mathbf{x}^+(k+1)\right\} \tag{18c}$$

$$\begin{aligned}\mathbf{p}_{J-1}(k\,|\,k;\mathbf{x}^+(k+1))&=Cov\left\{\mathbf{x}^-(k),x_{J-1}(k)\,|\,Y(k),\mathbf{x}^+(k+1)\right\}\\ &=Cov\left\{\hat{\mathbf{x}}^-(k\,|\,k;x_0(k),\mathbf{x}^+(k)),x_{J-1}(k)\,|\,Y(k),\mathbf{x}^+(k+1)\right\}\end{aligned} \tag{18d}$$

$$\begin{aligned}\mathbf{P}^-(k\,|\,k;\mathbf{x}^+(k+1))&=Cov\left\{\mathbf{x}^-(k)\,|\,Y(k),\mathbf{x}^+(k+1)\right\}\\ &=Cov\left\{\hat{\mathbf{x}}^-(k\,|\,k;x_0(k),\mathbf{x}^+(k))\,|\,Y(k),\mathbf{x}^+(k+1)\right\}\\ &\quad+E\left\{\mathbf{P}^-(k\,|\,k;x_0(k),\mathbf{x}^+(k))\,|\,Y(k),\mathbf{x}^+(k+1)\right\}\end{aligned} \tag{18e}$$

The quantities (18a) - (18e) and hence $\hat{\mathbf{x}}(k\,|\,k;\mathbf{x}^+(k+1))$ and $\mathbf{P}(k\,|\,k;\mathbf{x}^+(k+1))$ can be computed using (12) and (15) (note that $(x_0(k),\mathbf{x}^+(k))=(\mathbf{x}^+(k+1),x_{J-1}(k))$ so that $\hat{\mathbf{x}}^-(k\,|\,k;x_0(k),\mathbf{x}^+(k))$ = $\hat{\mathbf{x}}^-(k\,|\,k;\mathbf{x}^+(k+1),x_{J-1}(k))$ and $\mathbf{P}^-(k\,|\,k;x_0(k),\mathbf{x}^+(k))$ = $\mathbf{P}^-(k\,|\,k;\mathbf{x}^+(k+1),x_{J-1}(k))$, and so the expected values in (18) can in fact be computed by summing against $p(x_{J-1}(k)\,|\,Y(k),\mathbf{x}^+(k+1))$).

Summarizing, the measurement update for the BF consists of equations (17) and (16) for $\hat{\mathbf{x}}(k\,|\,k;\mathbf{x}^+(k+1))$, $\mathbf{P}(k\,|\,k;\mathbf{x}^+(k+1))$ and $p(\mathbf{x}^+(k+1)\,|\,Y(k))$, which are computed using equations (18), (12) and (15). The desired estimate $\hat{x}_K(k\,|\,k)$ is obtained from the last component of $\hat{\mathbf{x}}(k\,|\,k;\mathbf{x}^+(k+1))$ by summing against $p(\mathbf{x}^+(k+1)\,|\,Y(k))$.

Prediction Update for the Bayesian Filter

In this step we obtain the predicted estimate $\hat{\mathbf{x}}(k+1\,|\,k;\mathbf{x}^+(k+1))$ and the error covariance

$\mathbf{P}(k+1 \mid k;\mathbf{x}^{+}(k+1))$ in terms of the filtered estimate $\hat{\mathbf{x}}(k \mid k;\mathbf{x}^{+}(k+1))$ and error covariance $\mathbf{P}(k \mid k;\mathbf{x}^{+}(k+1))$. In view of the model (4) we have

$$\hat{\mathbf{x}}(k+1 \mid k;\mathbf{x}^{+}(k+1)) = \mathbf{A}\hat{\mathbf{x}}(k \mid k;\mathbf{x}^{+}(k+1)) + m_d\mathbf{e}_1 \tag{19a}$$

$$\mathbf{P}(k+1 \mid k;\mathbf{x}^{+}(k+1)) = \mathbf{A}\mathbf{P}(k \mid k;\mathbf{x}^{+}(k+1))\mathbf{A}^{H} + \sigma_d^2\mathbf{e}_1\mathbf{e}_1^{T} \tag{19b}$$

where $\mathbf{e}_1 = [1,0,...,0]^T$.

The BF is initialized by:

$$\hat{\mathbf{x}}(1 \mid 0;\mathbf{x}^{+}(1)) = E\{\mathbf{x}(1)\} = m_d\mathbf{1} \tag{20a}$$

$$\mathbf{P}(1 \mid 0;\mathbf{x}^{+}(1)) = Cov\{\mathbf{x}(1)\} = \sigma_d^2\mathbf{I} \tag{20b}$$

$$p(\mathbf{x}^{+}(1) \mid Y(0)) = p(\mathbf{x}^{+}(1)) = p_d(x_1(1)) \cdot \cdots \cdot p_d(x_{J-1}(1)) \tag{20c}$$

where $\mathbf{1}$ denotes a vector of 1's, and $\mathbf{I}$ is the identity matrix.

C. Computational Complexity and Adaptive Complexity Reduction

A major advantage of the BF is that it allows a tradeoff between performance and complexity by selecting a chip length J while retaining an adequate filter lag K. Table 1 shows the approximate complexity of the J chip BF (subsequently denoted by J-BF), as well as the DFE, KF and MLSE for comparison. The DFE is the simplest algorithm, followed by the KF, the 1-BF, the J-BF and the MLSE. The J-BF, ML sequence estimator, and MAP symbol estimator (not shown in the table) have complexity of order M^J, M^L, and $M^{\max\{K,L\}}$, respectively (and these exponential terms usually dominate the complexity for large M and/or L). Since we typically choose $J < \min\{K,L\}$, the BF decouples the exponential complexity from both the channel length and the filter lag.

Because of the nature of the statistics retained in the BF algorithm, a simple adaptive complexity reduction algorithm can be applied to the BF, resulting in what we call the reduced-state BF (RSBF). By comparing the posterior mass $p(\mathbf{x}^{+}(k+1) \mid Y(k))$ to a threshold p_t, and removing the state $\mathbf{x}^{+}(k+1)$ from the statistic if the value falls below the threshold, the size of the statistic may be adaptively reduced without losing important information. Let $1 \le M_t(k) \le M^{J-1}$ denote the number of the survivor states at time k. Then the computational complexity of the J-RSBF is of order $M \cdot M_t(k)$, compared with M^J for the J-BF. The simulation results in the next section show that the RSBF provides almost the same performance as the BF along with dramatic complexity reduction (especially at high SNR and with large M).

III. Simulation Results

The performance of the BF is characterized by Monte Carlo simulation with two discrete-time channel models. The two channel models are given by $\mathbf{c}^{(1)}(k) = [0.67, 0.47, 0.26, -0.52]$ and $\mathbf{c}^{(3)}(k) = [0.227, 0.406, 0.688, 0.406, 0.227]$. These are nonminimum phase channels with deep spectral nulls (channel 3 is taken from [11]). The coefficients of the channels are normalized so that the energy gain over the channels is unity.

We first examine channel 1 deconvolution with 4-QAM signalling. In Fig. 3 we show bit error probability (BEP) curves for J-BFs with different chip numbers J. For comparison the BEP curves of commonly used deconvolution techniques (e.g., MLSE (decoding delay = $5L$), DFE, KF) and ideal coherent demodulation (no ISI) are also presented. Each BEP curve is obtained from 500,000 Monte Carlo trials. Examination of Fig. 3 demonstrates some of the important characteristics of the various algorithms. Linear equalizers (e.g. the KF) do not generally provide the desired deconvolution performance. The DFE at low SNR has severely degraded performance compared with the linear equalizer; the DFE at high SNR obtains the desired exponential BEP curve but still has significantly degraded performance compared with MLSE. The BF at low SNR performs similarly to the linear equalizer; the BF at high SNR obtains the desired exponential BEP curve, and has performance approaching that of MLSE as J increases (note that most of the performance gain is achieved with $J = 3$). In Fig. 4 we show MSE learning curves for the BF and KF at E_b/N_0 = 15 dB. Each curve is obtained by ensemble averaging over 400 trials. It is seen that the BFs quickly reach steady-state (after 30-

40 iterations) and a much lower MSE than the KF.

We next examine the more difficult problem of channel 3 equalization with 16-QAM signalling. In Fig. 5 we show BEP curves for the 2-BF and the 3-RSBF for various values of the threshold p_t. For comparison the BEP curves for the DFE and ideal coherent demodulation (no ISI) are also presented (the MLSE performance is not shown because it is impractical to simulate the MLSE with 16^5 states). Examination of Fig. 5 shows that at high SNR the 2-BF has approximately a 3 dB better performance than the DFE, and at $E_b/N_0 = 22$ dB the 3-RSBF with $p_t = 10^{-7}$ has approximately a 3 dB better performance than the 2-BF. In general, the BEP curves of the RSBF's exhibit an error floor which depends on the threshold and above which the performance is similar to the BF; this allows the system designer to choose the (single) threshold in a principled fashion. In Fig. 6 we show an estimate of the steady-state mean value of $M_t(k)$, the number of survivor states retained by the RSBF. At $E_b/N_0 = 22$ dB the 3-RSBF with $p_t = 10^{-7}$ achieves roughly a factor of 35 reduction in complexity compared with the 3-BF. Viewed another way, the 3-RSBF provides about 3 dB better performance than the 2-BF at roughly half the complexity. Further simulation results for the BF and RSBF can be found in [8].

IV. Conclusions

This paper introduces a family of suboptimal fixed-lag smoothing filters called Bayesian filters (BFs). The family of BFs span the range between optimal linear filters and optimal symbol-by-symbol detectors in both performance and complexity. The BF is parameterized by a chip number J, with both performance and complexity increasing with J. The complexity of the BF is of order M^J, where M is the size of the signal constellation, and the chip number J is typically chosen smaller than both the filter lag and channel length.

The BF provides significant complexity reduction compared to the MLSE, and yet significant performance gains compared to traditional suboptimal deconvolution techniques (e.g., DFE) for relatively small J. An adaptive complexity reduction algorithm called the reduced-state Bayesian filter (RSBF) is proposed which produces dramatic reductions in complexity while still obtaining near optimum BEP performance.

References

[1] K. Abend and B.D. Fritchman, "Statistical Detection for Communication Channels with Intersymbol Interference," *Proc. IEEE*, vol. 58, May 1970, pp. 779-785.

[2] B.D.O. Anderson and J.B. Moore, *Optimal Filtering*, Prentice-Hall, 1979.

[3] A. Duel-Hallen and C. Heegard, "Delayed Decision-Feedback Sequence Estimation," *IEEE Trans. Commun.*, vol. COM-37, May 1989, pp. 428-436.

[4] V.M. Eyuboglu and S.U.H. Qureshi, "Reduced-State Sequence Estimation with Set Partitioning and Decision Feedback," *IEEE Trans. Commun.*, vol. COM-36, January 1988, pp. 13-20.

[5] G.D. Forney, "Maximum Likelihood Sequence Estimation of Digital Sequences in the Presence of Intersymbol Interference," *IEEE Trans. Inform. Theory*, IT-18, May 1972, pp. 363-378.

[6] G.D. Forney, "The Viterbi Algorithm," *Proc. IEEE*, 61, March 1973, pp. 268-278.

[7] R.E. Lawrence and H. Kaufman, "The Kalman Filter for the Equalization of a Digital Communications Channel," *IEEE Trans. Commun.*, vol. COM-19, December 1971, pp. 1137-1141.

[8] G.K. Lee, S.B. Gelfand, and M.P. Fitz, "Bayesian Techniques for Deconvolution," Technical Report, School of Electrical Engineering, Purdue University, West Lafayette, IN, 1993

[9] W.U. Lee and F.S. Hill, "A Maximum Likelihood Sequence Estimator with Decision-Feedback Equalization," *IEEE Trans. Commun.*, vol. COM-25, September 1977, pp. 1301-1327.

[10] C.J. Masreliez, "Approximate Non-Gaussian Filtering with Linear State and Observation Relations," *IEEE Trans. Automat. Contr.*, vol. AC-20, February 1975, pp. 107-110.

[11] J.G. Proakis, *Digital Communications*, McGraw-Hill, 1989.

[12] D. Williamson, R.A. Kennedy, and G.W. Pulford, "Block Decision Feedback Equalization," *IEEE Trans. Commun.*, vol. COM-40, February 1992, pp. 255-264.

Table 1. Complexity Comparison

ALGORITHM	MULTIPLICATION	ADDITION
J-BF	$[4(K-J+1)+1.5(K-J+1)^2]\cdot M^J$	$[3(K-J+1)+0.5(K-J+1)^2]\cdot M^J$
1-BF	$4K^2+5K+5M+L$	$2K^2+2K+4M+L$
KF	$4K^2+5K$	$2K^2+2K$
DFE	$L\cdot M$	$L\cdot M$
MLSE	$L\cdot M^L$	$L\cdot M^L$

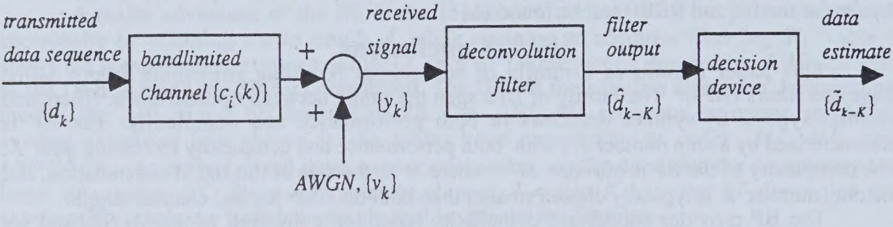

transmitted
data sequence
$\{d_k\}$
bandlimited
channel $\{c_i(k)\}$
+
+
received
signal
$\{y_k\}$
AWGN, $\{v_k\}$
deconvolution
filter
filter
output
$\{\hat{d}_{k-K}\}$
decision
device
data
estimate
$\{\tilde{d}_{k-K}\}$

Figure 1. Baseband model of a communication link

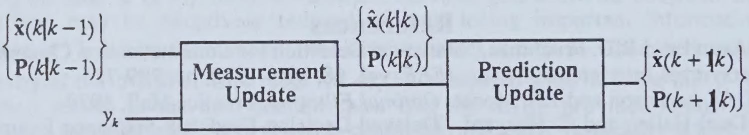

$\left\{\begin{matrix}\hat{\mathbf{x}}(k|k-1)\\ \mathbf{P}(k|k-1)\end{matrix}\right\}$
y_k
Measurement
Update
$\left\{\begin{matrix}\hat{\mathbf{x}}(k|k)\\ \mathbf{P}(k|k)\end{matrix}\right\}$
Prediction
Update
$\left\{\begin{matrix}\hat{\mathbf{x}}(k+1|k)\\ \mathbf{P}(k+1|k)\end{matrix}\right\}$

Figure 2(a). Recursion flow diagram for 1-BF.

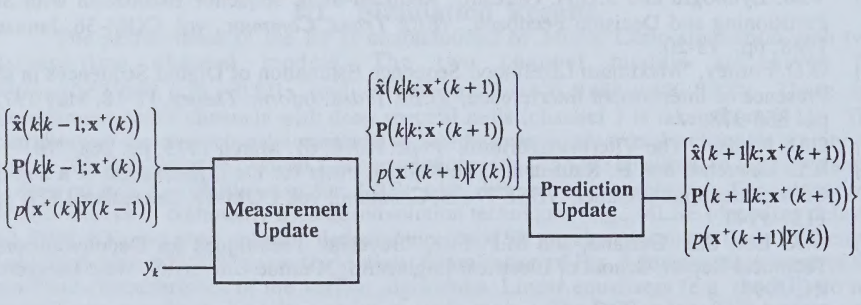

$\left\{\begin{matrix}\hat{\mathbf{x}}(k|k-1;\mathbf{x}^+(k))\\ \mathbf{P}(k|k-1;\mathbf{x}^+(k))\\ p(\mathbf{x}^+(k)|Y(k-1))\end{matrix}\right\}$
y_k
Measurement
Update
$\left\{\begin{matrix}\hat{\mathbf{x}}(k|k;\mathbf{x}^+(k+1))\\ \mathbf{P}(k|k;\mathbf{x}^+(k+1))\\ p(\mathbf{x}^+(k+1)|Y(k))\end{matrix}\right\}$
Prediction
Update
$\left\{\begin{matrix}\hat{\mathbf{x}}(k+1|k;\mathbf{x}^+(k+1))\\ \mathbf{P}(k+1|k;\mathbf{x}^+(k+1))\\ p(\mathbf{x}^+(k+1)|Y(k))\end{matrix}\right\}$

Figure 2(b). Recursion flow diagram for J-BF (J>1).

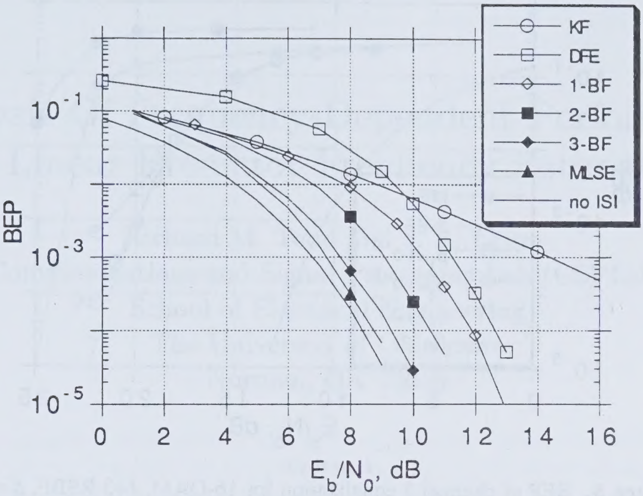


Figure 3. BEP of channel 1 equalization for 4-QAM signaling, *K*=7, 500,000 trials.

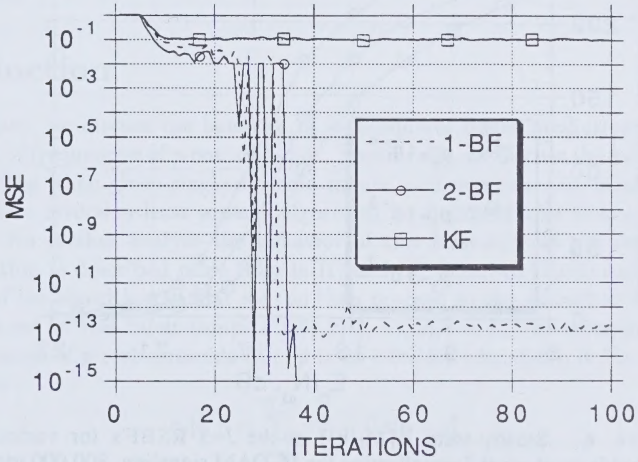


Figure 4. MSE of channel 1 equalization for 4-QAM signaling at E_b/N_0=15dB, *K* =7, 400 trials.

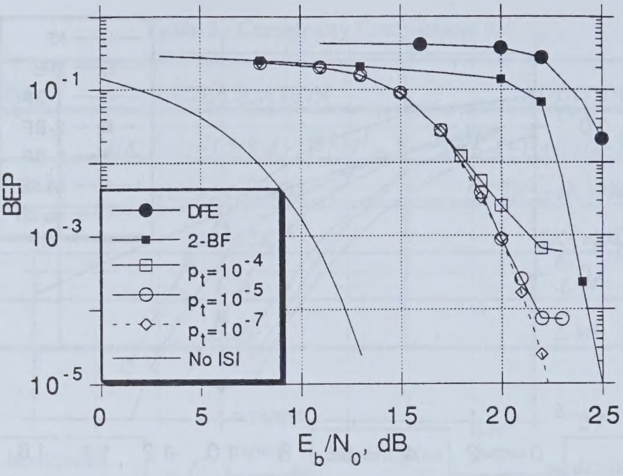


Figure 5. BEP of channel 3 equalization for 16-QAM, *J*=3 RSBF, *K*=7, 800,000 trials.

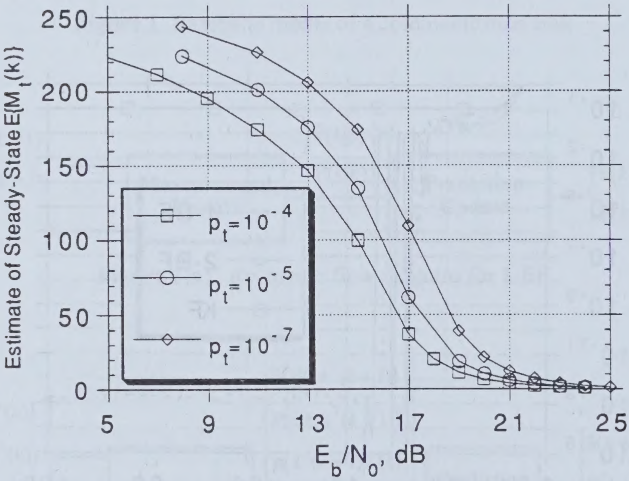


Figure 6. Steady-state $E\{M_t(k)\}$ of the *J*=3 RSBF's for various thresholds on channel 3 equalization for 16-QAM signaling, 800,000 trials.

Analysis Of Frequency-Dependent Performance Of Linear Predictor Frequency Estimators

Richard M. Todd and J. R. Cruz
Communications and Signal Processing Lab (CSPLab)
School of Electrical Engineering
The University of Oklahoma
Norman, OK 73019

Abstract

A fast least-squares algorithm for frequency estimation of real sinusoids is discussed. The results of simulations of the performance of this algorithm are given, and the large variations in the performance of the algorithm as a function of frequency are noted. Statistical analysis is presented to explain this variation, and it is shown that this frequency-dependent behavior is not specific to this algorithm, but seems to be dependent on the more general nature of this least-squares minimization problem.

Introduction

In this paper, we discuss the behavior of least-squares linear prediction algorithms for estimation of frequencies of p real sinusoids. Specifically, we discuss the behavior of a new fast algorithm when given single sinusoid inputs, and compare this behavior with that of the matrix inversion least-squares algorithm for handling this frequency estimation problem. We further analyse the behavior of this least-squares problem statistically. Our algorithm is described more fully in [1] and [2]; here, we briefly describe the basic formulae of the algorithm so that we can then proceed to the statistical analysis.

Given a set of real input samples $x[0], x[1], \ldots, x[N-1]$, the algorithm to estimate the frequencies of p real sinusoids (i.e., $\hat{p} = 2p$ complex sinusoids), it first computes the covariances

$$r_{\hat{p}}[i,j] = \sum_{n=\hat{p}}^{N-1} x[n-i]x[n-j] \qquad . \tag{1}$$

Next it computes the moments m_i defined as

$$m_i = \sum_{j=0}^{i} \binom{i}{j} r_{\hat{p}}[p, p+i-2j] \quad . \tag{2}$$

Then, it constructs two arrays q_i^j and e_i^j (the so-called "QD table") and initializes the first column as follows:

$$q_1^i = \frac{m_{i+1}}{m_i} \tag{3}$$

where e_p^i is set to 0, and then it computes all the other elements of the array via the so-called "rhombus rules"

$$\begin{aligned} q_i^j + e_{i-1}^j &= q_i^{j-1} + e_i^{j-1} \\ q_{i-1}^j e_{i-1}^j &= q_i^{j-1} e_{i-1}^{j-1} \quad . \end{aligned} \tag{4}$$

As a result of the theorems of Rutishauser [3] and additional results given in [2], it can be shown that the columns q_i^j converge, as $j \to \infty$, to the values

$$Q_i = \lim_{j \to \infty} q_i^j = 2\cos(2\pi f_i) \tag{5}$$

where f_i is the frequency of the ith sinusoid, scaled such that the Nyquist frequency is $f = 1/2$.

It is not obvious from the above that this algorithm is in fact a least squares algorithm, but as is shown in [2] or [4], the above Q_i values are indeed the roots of a polynomial, related to the forward predictor polynomial, which results in the minimum least squares predictor error.

Performance of the Algorithm

Here we present results of tests of this algorithm in the single sinusoid case. The test data sets were sets of 32 data points of the form

$$x[i] = \cos(2\pi f i) \qquad i = 0, \ldots, 31 \quad . \tag{6}$$

The normalized frequency f was varied between 0 and the Nyquist frequency; for each frequency, 1000 test runs were performed at each of 5 different signal-to-noise ratios (SNR) with noise added to the test data. For each combination of SNR and frequency (each set of 1000 test runs), the variances of the frequency estimates provided by the algorithm were computed.

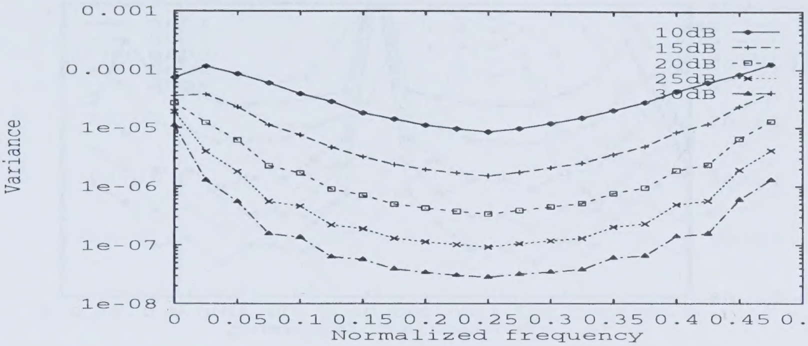


Fig. 1. Variance of frequency estimates, p=1 case.

Fig. 1 is a plot of the variance of the frequency estimates in the $p = 1$ case; similarly, Fig. 2 shows the $p = 2$ results and Fig. 3 shows those for $p = 3$. One notes from the plots that the variance of the frequency estimates in the $p = 2$ and $p = 3$ cases are unusually large for input signal frequencies in the region around $\omega = \pi/2$ (i.e, $f = 0.25$, or half the Nyquist frequency). This is a rather surprising phenomenon. At first, it was believed that this phenomenon was unique to this algorithm. To investigate this, we performed tests of what we called the matrix inversion least-squares algorithm. Basically, as shown in [2], our frequency estimation problem reduces to finding the roots Q_i of a (monic) polynomial $B_p(x)$ which has minimal norm with respect to the inner product over the polynomial ring $\mathbb{R}[x]$ defined as follows:

$$\langle f(x), g(x) \rangle = \sum_{i=0}^{p} \sum_{k=0}^{p} f[k] m_{ik} g[i] \tag{7}$$

where $f[i]$ and $g[i]$ are the coefficients of $f(x)$ and $g(x)$, respectively, and where m_{ik} is a matrix defined by

$$m_{ik} = \sum_{j=0}^{i} \sum_{l=0}^{k} \binom{i}{j} \binom{k}{l} r_{\hat{p}}[p + k - 2l, p + i - 2j] \quad . \tag{8}$$

(The norm of the polynomial $B_p(x)$ turns out to be just the total least- squares predictor error resulting from the predictor polynomial

$$A(z) = z^{-p} B_p(z + z^{-1}) \quad , \tag{9}$$

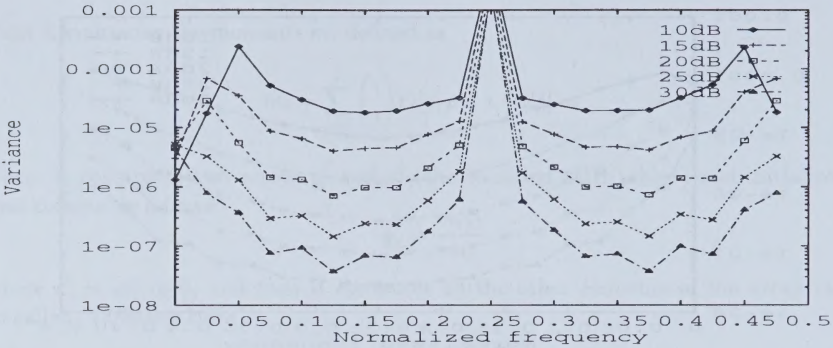


Fig. 2. Variance of frequency estimates, p=2 case.

since we are assuming our input signal is a combination of real sinusoids, the predictor polynomial must always be a function of $(z + z^{-1})$.) Note that the moment vector m_j defined at the beginning is just m_{j0}.

Our matrix inversion algorithm is, simply, to solve this minimization problem in the obvious way: first, compute the elements of the matrix m_{ik} from the input data. Then find the minimum-norm set of polynomial coefficients b_i by solving the matrix equation

$$m_{ik} b_k = -m_{ip} \qquad i, k = 0, \ldots, p-1 \qquad . \tag{10}$$

Then find the roots of the equation

$$x^p + b_{p-1} x^{p-1} + \ldots + b_1 x + b_0 = 0 \qquad . \tag{11}$$

These roots are the Q_i values we seek, and from them the individual frequencies can be derived.

Fig. 4 is a plot of the variance of the frequency estimates from this matrix inversion least-squares algorithm. As can be seen, the frequency estimates for this algorithm show the same behavior as those for the QD-based algorithm. So, it would appear that this behavior is not caused by something peculiar to the QD algorithm, but is somehow dependent on the nature of the minimization problem itself, since a matrix inversion solution of the minimization problem gives the same behavior. As we now show, statistical analysis can give some insight into this behavior of the minimization problem.

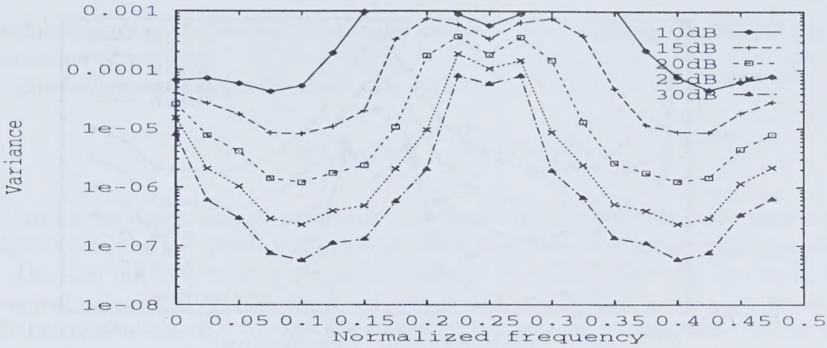


Fig. 3. Variance of frequency estimates, p=3 case.

Statistical Analysis

In our previous paper [1], we presented a partial statistical analysis of the QD-based algorithm in the simplest possible case (a single input sinusoid in white noise, with sample length $N \to \infty$). In this paper, we apply the same techniques to the analysis of the matrix inversion minimization algorithm.

Our signal is assumed to contain a single real sinusoid, i.e., $x[i]$ has the form

$$x[i] = A\cos(\omega i + \phi) \quad i = 0, \ldots, N-1 \quad . \tag{12}$$

In the limit as $N \to \infty$, the noise-free values of the covariances $r_{\hat{p}}[i,j]$ are

$$r_{\hat{p}}[i,j] \;=\; \frac{NA^2}{2}\cos(\omega(i-j)) \quad . \tag{13}$$

Let us write $Q = 2\cos(\omega)$. Then we can write

$$r_{\hat{p}}[i,j] \;=\; \frac{NA^2}{2}T_{|i-j|}(Q/2) \tag{14}$$

where $T_i(x)$ is the ith Chebyshev polynomial. In the presence of Gaussian white noise of variance σ^2 (assumed to be uncorrelated with the signal), the expected values of the covariances are perturbed as follows:

$$\tilde{r}_p[i,j] = r_p[i,j] + N\sigma^2\delta_{ij} = \frac{NA^2}{2}T_{|i-j|}(Q) + N\sigma^2\delta_{ij} \quad . \tag{15}$$

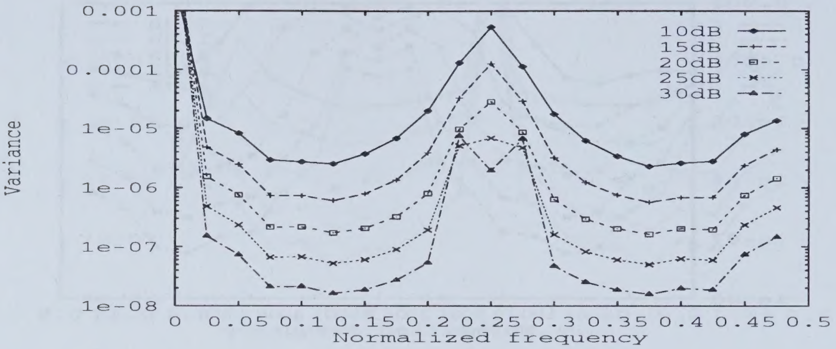


Fig. 4. Variance of frequency estimates from matrix inversion algorithm, p=2 case.

From this, one can compute the expected values of the elements of the array m_{ij}, and learn something about the behavior of the minimization problem in the single-sinusoid case.

For example, in the $p = 2$ case, the determinant of the matrix of the expected values of the m_ij is

$$\det(m_{ij}) = \left(\frac{NA^2}{2}\right)^2 \left(2s^2 + 2s^4 + s^2Q^2\right) \tag{16}$$

where $s^2 = \sigma^2/(A^2/2)$ is the inverse of the signal-to-noise ratio. Note that in the region where the normalized frequency is half the Nyquist freqency, Q is near zero; thus, the above determinant reaches a minimum. Thus, it is not at all surprising that the worst-case scenario for any attempt to solve our minimization problem is for the case when the frequency is around half the Nyquist frequency and Q is near zero.

Similarly, the determinant of the matrix for the $p = 3$ case is

$$\det(m_{ij}) = \left(\frac{NA^2}{2}\right)^3 \left(12s^4 - 6s^4Q^2 + 4s^6 + 2s^4Q^4\right) \quad . \tag{17}$$

Here, the minima are at $Q = \pm\sqrt{6}/2$, and while there is a local maximum around $Q = 0$, it is still considerably smaller than the higher values of the determinant in the regions around $Q = \pm 2$. Thus, it is not terribly surprising that there is a fairly broad range of impairment of the algorithm in the region from $Q = \sqrt{6}/2$ to $Q = -\sqrt{6}/2$ (corresponding approximately to $f \in [0.15, 0.35]$), which we observe. Also, there is a slight **decrease** in the variance of the frequency estimates around $Q = 0$, which corresponds to the local maximum of $\det(m_{ij})$.

Similarly, note that for the $p = 1$ case we have

$$\det(m_{ij}) = \left(\frac{NA^2}{2}\right)\left(1 + s^2\right) \tag{18}$$

which has no dependence on frequency; this leads one to expect the performance of the algorithm in the $p = 1$ case will not show much frequency dependence, as is the case.

One can not only look at the determinant of the matrix, but also the roots of the minimal polynomial (11), in the $p = 1$, $p = 2$, and $p = 3$ cases, given what we know of the behavior of the m_{ij}. For the $p = 1$ case, things are quite simple, and the root of the minimal polynomial is

$$\tilde{Q} = \frac{Q}{(1 + s^2)} \quad . \tag{19}$$

The error of this estimate of Q from the true value is $(Qs^2)/(1+s^2)$, which does not show a maximum magnitude at $Q = 0$ (in fact, the error is zero there), but does at $Q = \pm 2$. This fits nicely with the experimental observations.

For the $p = 2$ case we have the roots of the minimal polynomial as

$$\tilde{Q}_{1,2} = \frac{Q^3 - 2Q \pm \sqrt{Q^6 + 12Q^4 + 36Q^2 + 32s^2 + 32s^4 + 48s^2Q^2}}{(Q^2 + 2s^2 + 2)} \quad . \tag{20}$$

As expected, in the limit as $s \to 0$ one of the above roots approaches Q (the other root approaches $-4Q/(Q^2 + 2)$). A plot of the $\tilde{Q}_i$ versus Q and s is shown in Fig. 5. Figure 6 shows a closeup view of the difference between $\tilde{Q}_i$ and Q as a function of Q and s, showing that the error increases in the region around $Q = 0$, as expected. The results for the $p = 3$ case are similar, only with the larger errors occuring around $Q = \pm\sqrt{6}/2$, as one would expect from the determinant of the m_{ij} in this case. The expressions for the $\tilde{Q}_i$ are too complicated to present here (being roughly 60 lines worth of code in the Maple programming language), but the graphs of the $\tilde{Q}_i$ and the errors $\tilde{Q}_i - Q$ are shown in Figs. 7 and 8.

Conclusions

The QD-based algorithm [1],[2], and the matrix inversion least-squares algorithm have performance that varies significantly as a function of the frequency of the input signal in some cases (specifically, the over-determined cases of $p = 2$ and $p = 3$). This variation can be understood by looking at the mathematical expressions for the roots of the minimal polynomial given the known expected values of the m_{ij} matrix. Hence, it would appear likely that any least-squares algorithm for estimating the frequencies of real sinusoids will have problems with performance varying strongly as a function of frequency, especially in the over-determined cases.

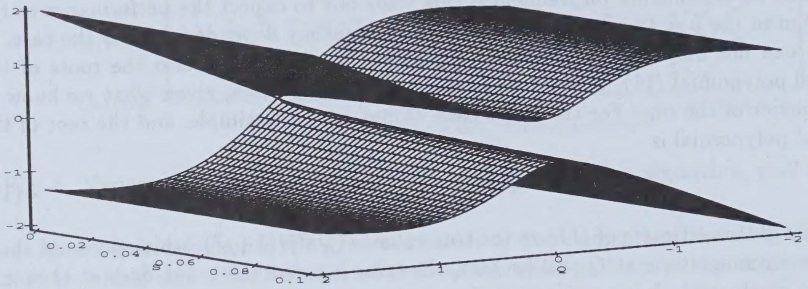


Fig. 5. Estimates of Q vs. Q and s, p=2 case.

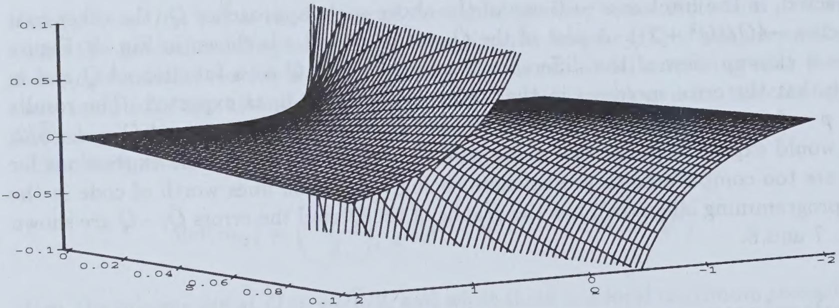


Fig. 6. Error of estimate of Q vs. Q and s, p=2 case.

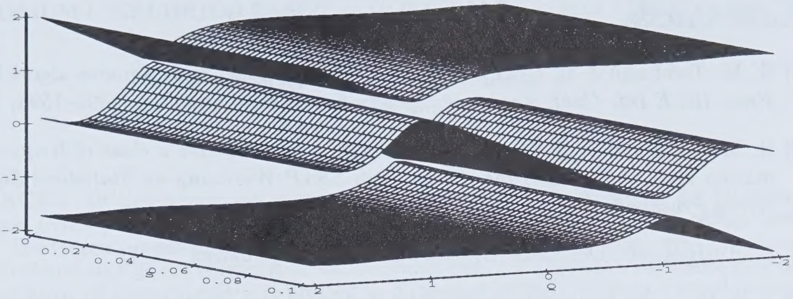


Fig. 7. Estimates of Q vs. Q and s, p=3 case.

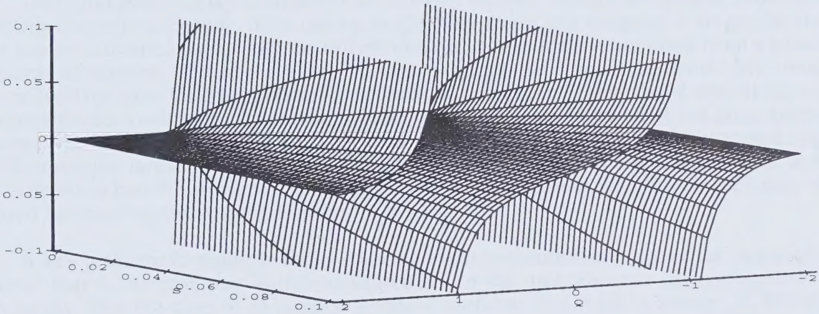


Fig. 8. Plot of error of estimate of Q vs. Q and s, p=3 case.

References

[1] R. M. Todd and J. R. Cruz, "A class of fast covariance least squares algorithms," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, pp. 1595–1598, 1990.

[2] R. M. Todd and J. R. Cruz, "A theoretical framework for a class of frequency estimation algorithms," in *Proc. IEEE Sixth SSAP Workshop on Statistical Signal and Array Processing*, pp. 400–403, 1992.

[3] H. Rutishauser, *Der Q–D Algorithmus.* Basel: Birkhauser, 1957.

[4] J. R. Cruz, "A general framework for a class of fast algorithms," in *Proc. Advances in Communications and Control Systems*, pp. 569–575, 1988.

MAXIMUM LIKELIHOOD BASED ALGORITHM FOR MULTIPLE TARGET TRACKING

A. Satish and Rangasami L. Kashyap
School of Electrical Engineering
Purdue University
West Lafayette, IN 47907

ABSTRACT : In this paper, we propose an algorithm (TAL) based on the Maximum Likelihood (ML) principle for multiple target tracking in near-field using sensor array outputs. The targets are assumed to be narrowband signals impinging on an array of sensors. The narrowband signals are modeled as sample functions of a Gaussian stochastic process. The phase delay of these signals is expressed as a function of both *range* and *bearing angle* of the respective targets. The range and angle parameters of all targets, referred to as *'track parameters'* are then simultaneously estimated. A *new* simplified likelihood function for the ML estimation of these parameters is derived using a *second* order approximation on the inverse of the covariance matrix of observations. The estimated track parameters are then utilized to track any arbitrary trajectory the targets describe in two dimensional space. Tracking is achieved by estimating the track parameters at regular time intervals wherein the targets would have moved to new positions in the neighborhood of their previous positions. Experimental results of TAL are presented and are compared with the methods of [5] and [9].

1. INTRODUCTION :

Time and phase-delay definitions are closely related, though estimation procedures can be significantly different. It is shown in [1] that, angle and range of a single far field source can be estimated using only the differential delay information obtained from a passive linear array of sensors. The observation is a vector of time-delay measurements. In contrast, the so called Direction Of Arrival (DOA) estimation problem for narrowband signals [2] uses signal magnitudes (voltage) as observations and phase delay information of the array outputs to estimate the bearing angles. In this paper, estimation of track parameters (range and angle only) for multiple targets in the near field situation is formulated in a manner similar to the DOA problem wherein the signals are considered to be Gaussian [2]. These estimates are then used for tracking the moving targets.

A key problem in multiple target tracking is the measurement/target data association problem : that is, determining which measurement in the measurement set corresponds to which target. For the case of D targets, standard methods like track splitting [3], PDA[4] etc., involve searching over the D! possible combinations. In sensor array based methods for target tracking, data association implies association of DOA estimates of different targets at two successive time instants.

Previous DOA based methods like the one proposed by Sword et. al [5] avoid this problem by deriving a recursive procedure for obtaining updated angle estimates at regular intervals of time whereas the method of Sastry et. al [6] uses knowledge of signal powers of the targets as additional information and necessarily requires that no two targets have the same signal power for correct data association. Both these methods use MUSIC [8] algorithm to obtain estimates of initial DOA and number of targets at regular time intervals.

Also, the above DOA based approaches do not use target dynamics for obtaining updated target positions in the sense that range and velocity of targets are not estimated. Angle estimates obtained at different time instants only provide directional information about the position of targets in space and no information about velocity. In the approach of [7], a dynamic model governing the motion of the targets is used. DOA estimates at each time interval are obtained using the 'stochastic ML' method for DOA estimation [15]. These DOA estimates are refined using a kalman filter wherein angular velocity and acceleration of the targets (components of the state vector in addition to the DOA) are also estimated.

Our method extracts additional *range* information along with DOA by using the *same* set of observations as in the above methods. With this, we are able to define the position of the target uniquely in a plane. The *power* of the proposed method comes from the formulation of a new likelihood function (in contrast to [7, 15]) which is maximized to get both DOA and range estimates of the D targets. Extraction of range information is made possible by expressing the phase delay as a function of *both* range and angle of the *respective* targets. Hence, data association, more specifically, the 'estimate association' problem is automatically solved by one-to-one association of the respective 2-tuples (range & angle estimates) of the targets at any two successive time instants. We can estimate the velocity also by using a kalman filter.

In Section 2, we formulate the problem using the stochastic Gaussian model for the signals (targets) and show how phase delay can be expressed as a function of the DOA and range parameters. In Section 3, we derive the new ML estimation procedure for the track parameters and discuss the TAL algorithm. Section 4 discusses implementation of TAL and comparison with two other methods [5] and [9]. Conclusions are included in the final section.

2. PROBLEM FORMULATION :

Consider a system of D moving targets to be tracked by a passive linear array of M sensors ($D < M$). The sensors are uniformly spaced and are separated by a distance 'd'. The targets can be assumed to move in a straight line at a constant velocity in the r-θ plane emitting narrowband plane waves impinging on the array from distinct directions $\theta_1, \theta_2, \ldots\ldots, \theta_D$. Note that this assumption is not necessary for the proposed algorithm but has only been used for simulation purposes. We assume the targets emit narrowband signals and hence the input signals, **x**(.) represented as complex envelopes are described as follows :

$$x_k(t) = \vartheta_k(t) e^{j\omega_k t} \quad k = 1, 2, \ldots, D \text{ and } t = 1, 2, \ldots, N. \tag{2.1}$$

(i) $\vartheta_k(t)$ is an independently identically distributed (i.i.d) Gaussian sequence $(0, \Gamma_{kk})$.

(ii) Two input signals $x_k(t)$ and $x_\ell(t)$ could be correlated,

$$E[\vartheta_k(t_1)\vartheta_\ell^*(t_2)] = 0 \text{ if } t_1 \neq t_2$$

since it is assumed that the signals are not correlated across time. The notation * is used to denote the complex conjugate of the quantity in question.

$$E[\vartheta_k(t_1)\vartheta_\ell^*(t_2)] = \Gamma_{k\ell} \text{ if } t_1 = t_2$$

since two signals observed at the same instant of time may be correlated.

Also, $E[x(t)x^*(t)] = \Gamma$ is the (DxD) covariance matrix of the signals.

If $\Gamma_{k\ell} = 0$, it is assumed that $\omega_k = \omega_\ell$.

The additive noise sequence $\{u(t) = 1, 2, \ldots, N\}$ is assumed to be i.i.d Gaussian $(0, \rho I)$.

The problem of estimating the track parameters, in a multi-target, multi-sensor array context can be reduced to estimating the parameters of the following model

$$y_m(t) = \sum_{k=1}^{k=D} f_{km} x_k(t) + u_m(t) \quad m = 1, 2, \ldots, M \text{ and } t = 1, 2, \ldots, N \tag{2.2}$$

N is referred to as the number of snapshots.

Let the vector function f(.) for the narrowband signals be

$$f_k = \text{col.}[1 \; e^{j\phi_{k2}} \; e^{j\phi_{k3}} \; \ldots\ldots \; e^{j\phi_{kM}}] \quad k = 1, 2, \ldots, D \tag{2.3}$$

where ϕ_{km} is the phase delay of the waveform from the kth target at the mth sensor.

We can show that [16] the phase delay ϕ_{km} can be expressed as a function of the bearing and angle parameters in the form

$$\phi_{km} = (m-1)(\pi \cos\theta_k) + (m-1)^2\left(\frac{\pi d \sin^2\theta_k}{r_k}\right) \tag{2.4}$$

where 'd' is the inter element array spacing.

Equation (2.4) is referred to as the Fresnel approximation [16]. This means that, in near field, the planar approximation is invalid, instead the wavefronts are modeled as quadratic surfaces.

Let $\psi_k = \pi\cos\theta_k$ and $\xi_k = \left(\dfrac{\pi d \sin^2\theta_k}{r_k}\right)$ k = 1, 2, ..., D. (2.5)

Substituting (2.5) in (2.4), we get the phase delay to be

$$\phi_{km} = \left[(m-1)\psi_k + (m-1)^2\xi_k\right] \quad k = 1, 2, ..., D \tag{2.6}$$

Substituting (2.6) in (2.3), we have the vectors f(.) to be

$$f_k = \text{col.}[1\ e^{j(\psi_k+\xi_k)}\ e^{j(2\psi_k+2^2\xi_k)}\ \ldots..\ e^{j((M-1)\psi_k+(M-1)^2\xi_k)}] \quad k = 1, 2, ..., D \tag{2.7}$$

Since f_k is a function of both ψ_k and ξ_k, k = 1, 2,.., D, we represent it in the form $f(\psi_k, \xi_k)$ for k = 1, 2, .., D. Substituting (2.7) in (2.2), we can write the model in vector - matrix format as

$$y(t) = A(\psi,\xi)x(t) + u(t) \qquad t = 1, 2, ..., N. \tag{2.8}$$

where y(t) is the M-vector of observations received by the M sensors. x(t) is the D-vector of the input signal, u(t) is the complex Gaussian additive noise vector mentioned previously. $A(\psi,\xi)$ is the MxD *'track'* matrix comprising of D transfer vectors and is given by

$$A(\psi,\xi) \equiv [f(\psi_1,\xi_1)\ f(\psi_2,\xi_2)\ \ldots\ f(\psi_D,\xi_D)] \tag{2.9}$$

ψ_k, ξ_k, k = 1, 2,..., D are the unknown track parameters to be estimated. These are our primary parameters of interest. ψ_k is a function of the arrival angle and ξ_k is the function of both the angle and the range of the kth target. The associated parameters like the signal powers and noise variance are not considered here, though they are also estimated simultaneously in the estimation problem.

3. DEVELOPMENT OF THE PROPOSED TRACKING ALGORITHM (TAL) :

3. 1. Maximum Likelihood Estimation of DOA and Range Parameters :

We state a theorem in which derive the proposed likelihood function for the estimation of target DOA, range and associated signal power and noise variance parameters.

THEOREM 1 :

The log likelihood of the observations {y(t), t = 1, ..., N} from D targets incident on the sensor array having M sensors is given by

$$\ln p(Y \mid \psi,\xi,\Gamma,\rho) = \frac{N}{2\rho}\frac{1}{M}\sum_{k=1}^{D} f_k^*\hat{R}f_k - \frac{N}{2\rho M^2}\sum_{k=1}^{D}\sum_{\ell=1}^{D} f_k^*\hat{R}f_\ell B_{k\ell} - \frac{N}{2}\text{tr}\left(\frac{B}{M}\right) + \frac{N}{4}\text{tr}\left(\frac{B}{M}\right)^2$$
$$+\ (\text{terms involving only } \Gamma,\rho)\ +(\text{terms not involving } \psi,\xi,\Gamma,\rho) + O\left(\frac{1}{M^3}\right). \tag{3.1}$$

where the 'B' matrix is defined to be $B_{\ell k} \equiv \left(\Gamma^{-1}\right)_{\ell k}\rho + (1 - \delta_{\ell k})f_\ell^* f_k$

$\delta_{\ell k} = 1, \qquad k = \ell$

$\delta_{\ell k} = 0, \qquad \text{else}$

The proof is in Appendix I.

We now give a two-step ML estimation procedure for the angle and range parameters of the D targets using a set of N snapshots.

Step 1: Obtain $(\hat{\psi}_k)_{ML}$ and $(\hat{\xi}_k)_{ML}$ by minimizing (3.1) with respect to ψ_k and ξ_k, k = 1, ..., D respectively.

Step 2: Using invariance property of ML estimator in (2.5), we get ML estimates of the DOA and range parameters to be

$$(\theta_k)_{ML} = \cos^{-1}\left[\frac{(\hat{\psi}_k)_{ML}}{\pi}\right] \tag{3.2}$$

and $$(\hat{r}_k)_{ML} = \left(\frac{\pi d \sin^2(\hat{\theta}_k)_{ML}}{(\hat{\xi}_k)_{ML}}\right) \tag{3.3}$$

which completes the estimation procedure.

3. 2. The Proposed Tracking ALgorithm (TAL) :

TAL comprises of the following steps :

(1). The inputs to the TAL algorithm at time t = nT are estimates of track parameters θ_k and r_k, k = 1, 2, .., D at time t = (n - 1)T and N snapshots at time t = nT.

(2). Use estimates of track parameters at the t = (n-1)T as initial guess values and estimate the angles and the ranges of the D targets using (3.2) & (3.3). This gives updated positions of the targets.

(3). Make a one-to-one association of the D 2-tuples (range and angle estimates) obtained at times t = nT and t = (n-1)T.

(4). Make (n = n + 1) and repeat steps (1)-(3) till end of tracking period.

Notes :

(a). The algorithm is started with intial guess values of θ_k and r_k, k = 1, 2, .., D (quantities to be estimated) in the neighborhood of their true values.

(b). The number of targets is always assumed to be known and is not estimated. Later on, we show that we can do reasonably well even without knowledge of the targets.

(c). The signal powers and noise variance if unknown can be estimated minimizing (3.1).

3. 2. 1. Features of proposed method :

(1). The criterion function maximized to obtain the track parameter estimates is different from that of 'stochastic ML' [7, 15]. In our method, the inverse of the data covariance matrix, R is approximated by a *second* order Taylor series expansion for large M, the number of sensors in the linear array. The resulting likelihood function is then maximized over all parameters of interest (DOA, signal powers and noise variance). This does *not* involve inversion of the MxM R matrix. Instead, we need to invert a DxD matrix, where D stands for the number of sources, when computing the B matrix. In practice, D<< M and this makes the algorithm computationally efficient.

(2). Both *range* and *angle* of the multiple targets are estimated using the sensor array outputs taking into account the curvature of the impinging wavefront. Hence, unique localization of targets on a plane is possible in near field.

(3). The proposed likelihood function is not symmetric in the components of the track parameters in view of large M approximation and hence the estimates we obtain are naturally ordered. This fact is true even if the signal and noise powers are unknown.

(4). Tracks are thus formed by simple one-to-one association of contiguous range and angle estimates.

(5). Though the proposed method is computationally efficient, it requires multidimensional non-linear optimization for estimating the track parameters.

(6). If the track parameter estimates are not in the neighborhood of their true values, error in estimation may propagate along the track. This is because, estimates obtained at a particular instant serve as initial guess values for estimating the parameters at the succeeding instant.

(7). TAL is iterative and information is processed in 'blocks' i.e., we get a set of snapshots in a small time interval, estimate the track parameters and these estimates (information) are utilized as initial guess values along with the next block of observations for estimating the new track parameters.

3. 2. 2. Comparison of TAL with other methods :

(1). Bearing and range estimates are extracted in contrast to only DOA estimates in [5-7].

(2). The method in [5] essentially involves least squares fitting and since our method is based on the ML principle, we obtain better performance (see simulation results).

(3). Velocity can also be estimated unlike in [5] and [6] using a kalman filter formulation.

(4). The method in [7] uses the well known stochastic ML method to estimate the DOAs and track the targets. As already mentioned in section (3.2.1), the proposed ML method involves computing the inverse of only a DxD matrix (D = number of targets) and hence is computationally efficient.

(5). If all the parameters of interest (DOA, signal powers and noise variance) are unknown and are estimated using 'stochastic ML', we will still have the problem of data association [7].

This is because (I.2) is symmetric in the components of the DOA vector. Hence, in [7], an alternate likelihood function is proposed wherein signal powers and noise variance are *known* and the signal powers are required to be necessarily different for identifiability of the different targets (for correct association). In contrast, the proposed likelihood function is *not* symmetric in the components of the track parameters. This is in view of the large M approximation and the joint representation of range and DOA of a particular target in the same transfer vector. Hence the estimates we obtain are naturally ordered. This fact is true even if the signal and noise powers are *unknown*.

(6). The individual angle and range estimates are better than the method of [9] wherein ESPRIT [11] is used in conjunction with the spatial version of the Wigner-Ville distribution.

(7). TAL, being an ML method gives good estimates for small N & low SNR.

4. PERFORMANCE OF THE PROPOSED METHOD :

We consider a situation with two moving targets. For simulation purposes, tracks of targets are generated assuming straight line motion with constant velocity though no such assumption is required for the algorithm. Different types of target tracks (vertical and crossing targets) are synthesized. It is important to note that data are generated according to the model equations (2.7) and (2.8). The true values of the track parameters are obtained from the synthesized tracks and are used to generate the snapshots. We calculate statistics of the estimated parameters like root mean-square error and evaluate the performance.

To start with, the TAL algorithm will be compared and contrasted with Swindlehurst and Kailath's method [9] of estimating the DOA and range parameters for near-field sources. Though the method in [9] just involves estimation of DOA and range and is not used for tracking purposes, a comparison is useful because our method relies on simultaneous estimation of angle and distance for target tracking.

4.1. Comparison with Swindlehurst and Kailath's Method :

The method in [9] uses a spatial analog of the Wigner-Ville distribution. for estimating the DOA and range of a near-field sources. A signal subspace method, ESPRIT [11] is employed for estimating the Wigner-Ville kernel frequencies.

The advantages in [9] is that, the source locations are estimated more accurately, closely-spaced sources are easily resolved and no search of a spectral surface is required. We show that using TAL, we can estimate the DOAs equally well or better and superior range estimates are obtained. We have simulated TAL for all their test cases but have directly taken their experimental results for comparison. An added advantage is that, our method is valid even for correlated targets and for an arbitrary array configuration. But being a ML method, it does involve multidimensional non-linear maximization. This problem can be alleviated in a dense target environment using Expectation Maximization-ML which decouples the high-dimensional optimization problem into smaller two dimensional optimizations [12] at the expense of more computation time.

For all simulations shown, we assume a uniform linear array of 16 sensors with inter element spacing $d = \lambda / 4$ (adequate for preventing spatial aliasing). Estimates are obtained after averaging over 100 runs of the algorithm. We consider the two target case, wherein the range and angle estimates are plotted versus the number of snapshots. The signal-to-noise ratio (SNR) chosen is 20 dB. We obtain good DOA estimates for both methods in all cases as shown in Fig. 1. However, the range estimates in [9] deteriorate for small number of snapshots. At close range (Fig. 2), it is seen that TAL gives good estimates whereas the range estimates get mixed up in the case of [9]. If we were to use this result for target tracking, it would result in wrong association of estimates. Even for small range difference of the order of 5λ (close range), TAL is able to distinguish the targets correctly and gives good estimates.

4. 2. Vertically Moving Targets and Comparison with Sword's Algorithm :

We track trajectories of targets moving in a straight line with constant velocity using the proposed TAL algorithm. In all our simulations, we have restricted ourselves to the case of two targets in order to study and interpret various factors more effectively. To start with, situations in which the targets do not cross each other are considered. We compare our method with Sword's algorithm [5] which presents results only for non - crossing targets moving in a straight line parallel to the array. In reality, the scenario considered could represent two aircraft on a sortie or surveillance mission. Approaching targets are considered because the radar cross section of any approaching target is minimal when compared to that of a target moving parallel to the tracking radar.

As already mentioned, we get only angle tracks from [5] since range is not estimated. But, in our method, using the same number of observations, in addition to DOA, we extract more information (range) about the targets. A comparison is made between the two methods to see the effectiveness of this new information to obtain good track estimates. The angle estimates obtained from both methods are compared. The estimates of [5] used for comparison are obtained by simulating the algorithm in [5]. We use ML method for getting estimates of the initial DOAs.

The results of comparison are shown in Figs. 3. and 4. From the angle tracks in Fig. 3, the TAL DOA estimates are closer to the true values than the DOA estimates in [5]. More importantly, as time increases, the RMSE in TAL shows a decreasing trend whereas it continuously increases in Sword's method. The log root mean-square error (RMSE) of the DOA estimates averaged over 20 runs is plotted versus the track period in Fig. 4. This shows that error in the estimates using Sword's method gets propagated as we move along the track. Stable estimates are not obtained from Sword's algorithm probably because of the Taylor's series expansion and the subsequent first order approximation of the transfer vector function. Moreover, it uses information of only the first column of the data covariance matrix. The differentiation operation introduces noise into the system which gets propagated. Though our method is iterative and is dependent on the accuracy of the ML estimates, it is more stable than Sword's method. It should be mentioned that errors can also propagate in TAL if the ML estimates at time $t = nT$, which are fed as initial guesses to the estimation procedure at time $t = (n+1)T$, are far away from the true values.

4. 3. Crossing Target Scenario :

We now consider the more realistic situation wherein the targets cross each other. The data association problem becomes crucial for this case. With various experimental results, we show that TAL is able to track the targets by correctly associating track parameter estimates over the duration of the track. Results are given both for targets moving with constant velocity in a straight line.

Experiment 1 (Two targets with constant velocity, high SNR and small sampling interval)

Two fast moving targets are considered with $v = 200$ m/s. The targets are assumed to be moving in a straight line woth constant velocities. The sampling interval is small (0.11s) and the SNR is high (10 dB) implying favorable operating conditions. The true and the estimated tracks of the TAL algorithm are plotted as shown in Fig. 5. The plotted track is the average of two test runs. The TAL algorithm is able to follow the targets' true trajectory over the entire tracking period accurately. It is also interesting to note that the errors increase in track (2) near the end of the track period. This can be explained by looking at the position of the target at that point in time. We see from Fig. 5 that the target is close to endfire and going away from the broadside array. This explains the drop in performance at low-angles.

Experiment 2 (Two targets moving constant velocity, low SNR and large sampling interval)

Experiment 2 illustrates the fact that the SNR and sampling interval are very important factors in correct tracking and association. We have the very same trajectories as in experiment 1, but decrease the SNR to 5 dB and double the sampling interval. The resulting target tracks plotted in the spatial domain in Fig. 6 show that the TAL method is not able to follow the correct track after cross-over. Reducing SNR is reduced (5 dB) and increasing sampling interval causes the following problem : the track parameter estimates are not close to their true values at cross over. This means for estimating the track parameters at the following time instant, these erroneous estimates which are used as initial guess values are not in the neighborhood of the true values. This causes further deterioration in the TAL estimates and thus the algorithm follows the wrong target track. A large increase in error after cross-over occurs since the targets tracks are flipped over.

It is thus seen SNR, velocity and the sampling interval are the important factors affecting the performance of the TAL method. The method performs well in a variety of situations as indicated by simulations and can be made feasible for practical implementation.

In [13], expressions are derived for the CRB on the bearing and range of a Gaussian signal source observed by a two-dimensional array in the presence of strong Gaussian interference. We have derived asymptotic CR bounds for the range and DOA estimates based on the large M Taylor's series expansion which will be presented in another paper. The family of curves for different parameters like N, M, SNR, DOA, and range suggest possible regions of operation of the tracking system for obtaining good estimates.

4.4. Extensions :

(1). Velocity can also be estimated using the Kalman filter formulation.

(2). Using estimated angle, range and velocity, one-step ahead prediction of position of targets can be achieved. The predicted angle and range estimates can be used as initial guesses for the ML estimation procedure to get better estimates of trajectories. But in our simulations, instead of the predicted estimates, the estimates themselves are used as initial guess values.

5. CONCLUSIONS:

A maximum likelihood algorithm (TAL) for estimating track parameters has been developed and its workability is demonstrated. It is shown that performance of TAL was superior to methods of [5] and [9]. Also, this problem of joint DOA and range estimation and tracking can be effectively used in mobile communications [14] where accurate tracking of the mobile can help in efficient utilization of the available radio spectrum.

REFERENCES

[1]. E. Weinstein, "Optimal Source Localization and Tracking from Passive Array Measurements," *IEEE Trans. ASSP.*, **30**, No. 1, 1982, 69-76.

[2]. A. Satish and R. L. Kashyap, "Maximum Likelihood Estimation and Cramer-Rao Bounds for Direction of Arrival Parameters with Narrowband Signals," *Submitted to IEEE Trans. Signal Processing.*, 1992.

[3]. P. Smith and G. Buechler, "A Branching Algorithm for Discriminating and Detecting Multiple Objects, " *IEEE Trans. Automatic Control.* **20**, No. 2, 1975, 101-104.

[4]. Y. Bar-Shalom and E. Tse, "Tracking in a Cluttered Environment with Probabilistic Data Association, " *Automatica.* **11**, 1975, 451-460.

[5]. C. K. Sword, M. Simaan, and E. W. Kamen, "Multiple Target Angle Tracking using Sensor Array Outputs," *IEEE Trans. AES.*, **26**, No. 2, 1990, 367-373.

[6]. C. R. Sastry, E.W. Kamen, and M. Simaan, "An efficient Algorithm for Tracking Angles of Arrival of Moving Targets, " *IEEE Trans. SP.* **39**, No. 1, 1991, 242-246.

[7]. C. R. Rao, C. R. Sastry, and B. Zhou, "Tracking the Direction of Arrival of Multiple Moving Targets, " *IEEE Trans. Signal Processing.*, accepted for publication.

[8]. R. O. Schmidt, "Multiple Emitter Location and Signal Parameter Estimation," *IEEE Trans. Antennas and Propagation.* **AP-34**, No. 4, 1986, 276-280.

[9]. A. L. Swindlehurst and T. Kailath, "Passive Direction-of-Arrival and Range Estimation for Near-Field Sources," in *ICASSP,* 1988, pp. 123-128.

[10]. C. R. Rao, *Linear Statistical Inference and its Applications*, John Wiley, NY, 1968.

[11]. R. Roy and T. Kailath, "ESPRIT-Estimation of Signal Parameters via Rotational Invariance Techniques, " *IEEE Trans. ASSP.*, **37**, No. 7, 1989, 984-995.

[12]. M. I. Miller & D. R. Fuhrmann, "Maximum Likelihood Direction finding of Stochastic Sources: A Separable solution," *IEEE Trans. ASSP.*, **38**, No. 9, 1990, 1560-1577.

[13]. H. Messer, Y. Rockah and P. M. Schultheiss, "Localization in the Presence of Coherent Interference," *IEEE Trans. ASSP.*, **38**, No. 12, 1990, 2025-2032.

[14]. Soren Anderson, Mille Millnert, Mats Viberg & Bo Wahlberg, "An Adaptive Array for Mobile Communication Systems," IEEE Trans. on Vehicular Technology, **40**, No. 1, 1991, 230-236.

[15]. P. Stoica and A. Nehorai, "Performance Study of Conditional and Unconditional Direction of Arrival Estimation," *IEEE Trans. ASSP.*, **38**, No. 10, 1990, 1783-1795.

[16]. B. D. Steinberg, *Principles of Aperture and Array System Design*, John Wiley & Sons, 1976.

APPENDIX I

Proof of Theorem 1

The proof of this theorem is similar to the derivation of the log likelihood function in [2] since the range information is included within the phase delay expression. We give only a brief exposition of it here. The covariance matrix of the observation vector y(t) is given by,

$$R \equiv E[y(t)y^*(t)] = \rho I + A(\psi,\xi)\Gamma A^*(\psi,\xi) \qquad (I.1)$$

R is an MxM matrix where M denotes number of sensors and $A(\psi,\xi)$ is defined as in (2.9). Since {y(t), t = 1,....,N} is an independent Gaussian sequence with zero mean and covariance R, the joint probability density of {y(t), t = 1,....,N} given ψ, ξ, Γ, ρ can be written as follows:

$$Y = \{y(t),\ t = 1,2,....,N\}$$

$$p(Y|\psi,\xi,\Gamma,\rho) = \prod_{t=1}^{N} 2\pi^{-M/2}(\det R)^{-1/2}\exp\left[-\frac{1}{2}y^*(t)R^{-1}y(t)\right]$$

$$= 2\pi^{-MN/2}(\det R)^{-N/2}\exp\left[-\frac{1}{2}\operatorname{trace}\left\{R^{-1}\sum_{t=1}^{N}y(t)y^*(t)\right\}\right]$$

Thus, the density possesses a sufficient statistic $\widehat{R}$ given by

$$\hat{R} = \frac{1}{N}\sum_{t=1}^{N}y(t)y^*(t)$$

Consequently, the log likelihood can be simplified as follows:

$$L(\psi,\xi,\Gamma,\rho) = -\ln p(Y|\psi,\xi,\Gamma,\rho) = \frac{MN\ln 2\pi}{2} + \frac{N}{2}J(\psi,\xi,\Gamma,\rho) \qquad (I.2)$$

where $J(\psi,\xi,\Gamma,\rho) = \ln(\det R(\psi,\xi,\Gamma,\rho)) + \operatorname{trace}(R^{-1}\hat{R})$

The key to simplify (I.2) is to first use the matrix inversion lemma [10] to express R^{-1} as

$$R^{-1} = \rho^{-1}I - A\left(A^*A + \rho\Gamma^{-1}\right)^{-1}A^*\rho^{-1} \qquad (I.3)$$

A second order Taylor's series approximation on R^{-1} is given by

$$R^{-1} = \left(\frac{1}{\rho}\right)\left[I - \left(\frac{AA^*}{M}\right) + \left(\frac{ABA^*}{M^2}\right)\right] + O\left(\frac{1}{M^3}\right) \qquad (I.4)$$

Equation (I.2) can be simplified using (I.4) to give

$$\ln p(Y \mid \psi,\xi,\Gamma,\rho) = \frac{N}{2\rho}\,\frac{1}{M}\sum_{k=1}^{D}f_k^*\hat{R}f_k - \frac{N}{2\rho M^2}\sum_{k=1}^{D}\sum_{\ell=1}^{D}f_k^*\hat{R}f_\ell B_{k\ell} - \frac{N}{2}\operatorname{tr}\left(\frac{B}{M}\right) + \frac{N}{4}\operatorname{tr}\left(\frac{B}{M}\right)^2$$

$$+\ (\text{terms involving only } \Gamma,\rho)\ +(\text{terms not involving } \psi,\xi,\Gamma,\rho) + O\left(\frac{1}{M^3}\right)$$

(I.5) (QED)

ACKNOWLEDGMENT

This work was supported by the Innovation Science and Technology (IST) Program of the BMDO monitored by the Office of Naval Research under contract ONR N00014-91-J-4126.

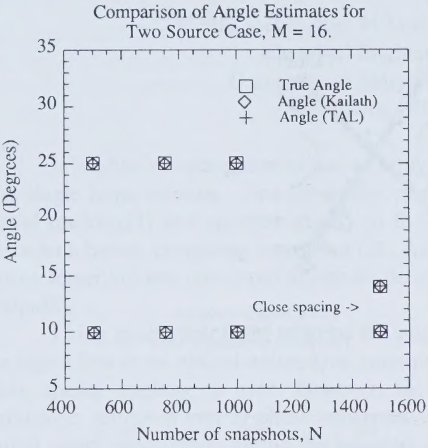


Fig. 1 : Angle estimates for two source case. Runs = 100. SNR = 20 dB.

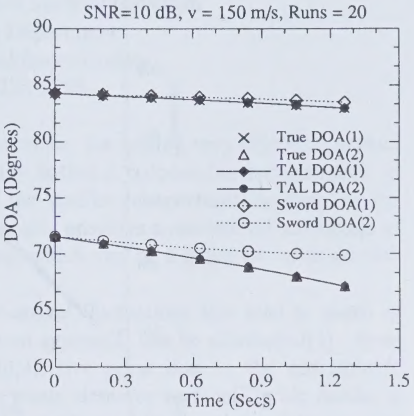


Fig. 3 : Averaged DOA tracks of the two targets for TAL and Sword's Methods.

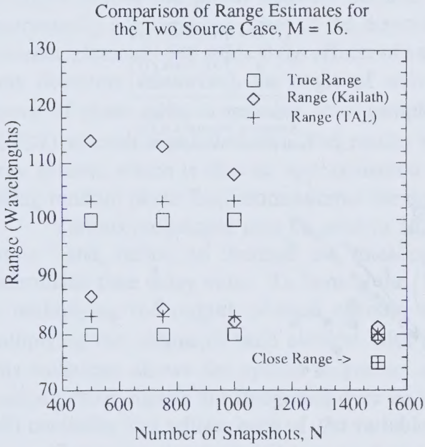


Fig. 2 : Range estimates for two source case. Runs = 100. SNR = 20 dB

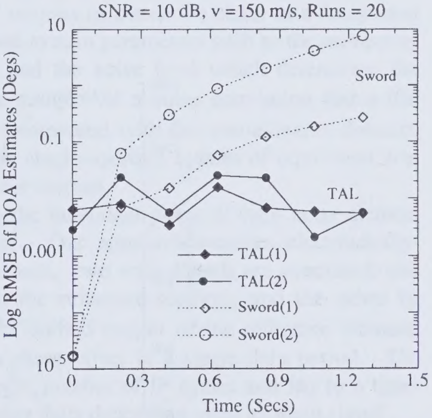


Fig. 4 : Log RMSE of DOA Estimates for both TAL and Sword's methods.

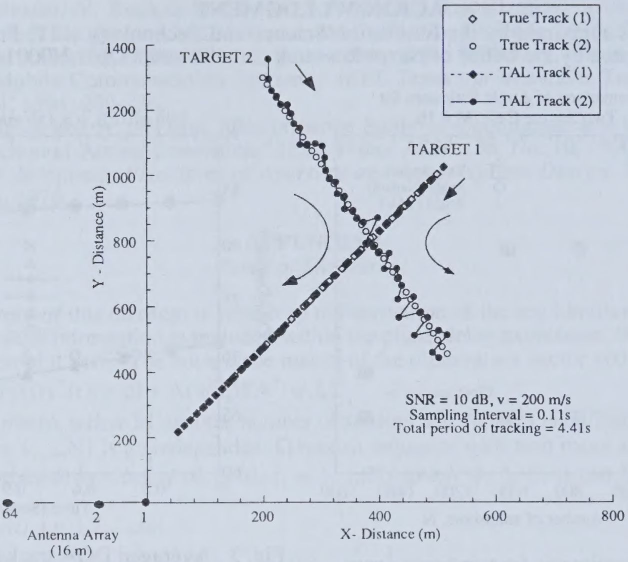


Fig. 5 : Both targets are moving towards the sensor array as indicated by the arrows. The dark dots show estimated tracks and the light ones show the true tracks.

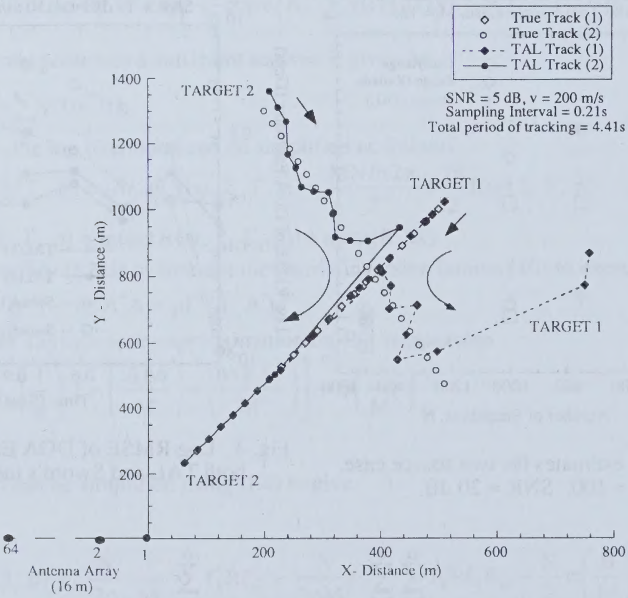


Fig. 6 : Target track plots for reduced SNR (5 dB). The TAL algorithm fails and the targets follow the wrong track as shown in the figure. Also, track parameter estimates are not as accurate as before.

ON ADAPTIVE PHASE ALIGNMENT OF AN APERTURE ARRAY FOR OPTICAL SATELLITE-LINK COMMUNICATIONS

Mehul Mehta, Joel M Morris, and Michael Buckman
Electrical Engineering Department
University of Maryland Baltimore County
Baltimore, MD 21228-5398

It is often advantageous to use an array of antennae for getting very high gain instead of a single large antenna. One advantage of an array is that it reduces the requirements on spatial tracking[1] and aperture quality to those of the smaller subapertures by adapting the phase and, hence, correcting wavefront tilt. An array also provides a mechanical advantage of having lesser volume compared to one single antenna, which may be a major factor in satellite design[4].

Using spatial tracking, most of the angle-of-arrival fluctuations that lead to much of the signal loss in an optical-heterodyne communication system[2] can be eliminated[1]. Even when spatial tracking is used, however, the multiplicative noise due to the atmospheric turbulence causes a loss of phase coherence at the photo detector surface[2]; this results in partial signal cancellation during the integration that is performed by the photo detector over its receiving surface.

We investigated an adaptive phase alignment technique that compensates for phase variations across the photo detector surface using a linear array of photo detectors, and electronically adjusting the phase of the detectors' outputs to match the phase of a designated reference element. We studied the effects of various system parameters such as the number of array detectors (elements), the angle of arrival, and the noise level which determines the degree of phase noise correlation from sample to sample via a noise correlation factor ($0 \leq NCF \leq 1$) at each array element. The results are compared with the non-adaptive detector array system, which is also an approximation to a single-aperture system of equivalent size having random phase fluctuations across the detector surface.

Various techniques may be used to adjust the individual phase of each array element output and, hence, to increase the total output. Our approach assumes electronically-controllable time delay units. To control the feedback, two error signals are generated: one by multiplying the output of each element with the reference element, and the other by multiplying the output of each element with a 90°-shifted output of the reference element. This technique allows the system to correct any phase offset in a single data period. The resulting error signals are integrated over an integer number of IF cycles and fed to a time-shift controller that adjusts each of the variable delay units depending upon its input signal.

Computer simulation[5] was used to evaluate the adaptive phase alignment system's performance, under the assumption that the data rate is 1 GHz, IF is 16 GHz, the individual aperture is a 5.1 mm square, the information sequence is randomly generated, and BPSK signaling[3] is used. The local oscillator output is assumed sufficiently high that the system is shot-noise limited with an uniform distribution. Satellite platform instability is modeled by harmonic motion of the optical components with respect to the incoming beam. The simulation trials are performed at decreasing NCF values, at varying number of array elements, and at varying signal angle-of-arrival values.

The results show that the average output power increases substantially when this

---

Work supported by NASA grant # NAG 5-2194

adaptive phase alignment technique is used to improve the coherence of the array signals before addition. For NCF=1.0 (i.e., the path length to each element may be different but fixed), the system output increases linearly with the number of elements (equivalently, as the aperture area increases). As NCF decreases, the rate of increase of the total output with the number of elements decreases. It is observed that this system gives robust SNR vs. angle-of-arrival performance for the given field of view compared to the non-adaptive system of equivalent area. Our results compare well with Mercer's experimental findings[4], which was based on electronic phase correction using a 4-element linear array. The phase corrected output in both studies fall within 1 dB of the ideal output (Fig. 1) occurring when all phases are perfectly matched (coherent sum). However, the spread of the measured output power vs. angle-of-arrival for the non-adaptive system is narrower then in Mercer's findings. This difference, we believe, is due to our assumption that each optical beam has an uniform intensity profile across each detector surface, whereas in Mercer's experiment the profile was a truncated Gaussian. This effect is equivalent to a reduction in the effective area of an aperture and, hence, a broadening of the spread of the SNR vs. angle-of-arrival profile.

In another simulation, we studied the effect of increasing the number of elements, but maintaining constant total aperture area of 5.1x20.4 mm^2. Even though it is assumed that the phase noise in all the elements are mutually independent, the results show very good improvement in SNR vs. angle-of-arrival performance over a wide range of angle-of-arrival (Fig 2). As NCF decreases for the adaptive system (from Fig 2(a) to Fig 2(b)): (1) the 4-element array's power vs. angle-of-arrival profile has slightly less coverage with respect to angle-of-arrival, and (2) the 16-element array's power vs. angle-of-arrival profile maintains the same broader coverage but at a reduced relative power level. In practice, the tradeoff in choosing the number of array elements may be between the total output power and robustness of the SNR vs. angle-of-arrival profile.

REFERENCES

[1] Bernays, D. J. and E. P. Colagiuri-Cafareli, " An Experimental Spatial Acquisition and Tracking system for Optical Intersatellite Crosslinks," *Lincoln Laboratory Technical Report* 923, 3 December 1991.

[2] Winick, K. A., " Atmospheric Turbulence-Induced Signals Fades on Optical Heterodyne Communication Links," *Applied Optics*, Vol. 25, No. 11, 1 June 1986.

[3] Cooper, G. R. and Clare D. M., *Modern Communication and Spread Spectrum*, McGraw-Hill, 1986.

[4] Mercer, L. B. , "Adaptive Coherent Optical Receiver Array," *Electronics Letters*, Vol. 26, No. 18, 30 August 1990.

[5] Mehta, M., " A Study of Adaptive Phase Alignment for an Aperture Array for Optical Satellite-Link Communications," *MSEE Thesis*, Spring 1994, University of Maryland Baltimore County (UMBC).

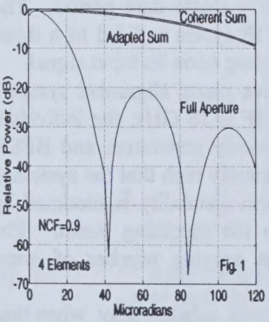


Fig. 1

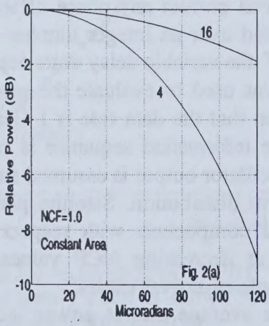


Fig. 2(a)

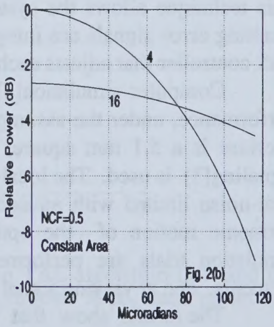


Fig. 2(b)

Recovering Extruded Surfaces from a Single Image

D.A. Forsyth[1] and C.A. Rothwell[2]

Abstract

Extruded surfaces consist of a section of general cone cut by two planes. Such surfaces possess a simple geometry, yet are widespread in the real world. The geometry of an extruded surface can be recovered from a single uncalibrated image. This geometric information can be fleshed out with the surface markings of the object, leading to a highly realistic 3D representation, which can be used for recognition or for rendering. We show examples based on images of real scenes.

1 Introduction

Efficient object recognition programs require distinctive object representations; in many applications, such as surveillance, video processing and image databases, only a single image is available. This paper shows how to recover an object representation that captures both shape and pattern for a large class of surfaces covering many man-made objects.

1.1 Recognition using indexing

In typical modern systems, recognition proceeds by:

- **Feature extraction:** Edges are extracted and the resulting geometric primitives are grouped into likely object groups (for example, a group of five lines, of two conics, or of a single "M-curve" in [11]).
- **Indexing and hypothesis merging:** Feature groups are used to compute geometric primitives to index the object in a model base (for example, the projective invariants of a pair of conics in [11]). Normally, one object leads to several groups of features indexing the appropriate model, so the resulting hypotheses must be merged using consistency criteria to obtain a single recognition hypothesis.
- **Verification:** Hypothesized objects are back-projected into the image; the results are used to obtain further information that may confirm the hypothesis.

Systems of this sort have been demonstrated for plane objects in a number of papers [3, 11, 4, 7, 13, 16]. Typically, object models consist of a system of invariant values and are therefore relatively sparse, meaning that hypothesis verification is required to confirm a model match. However, no searching of the model base is required because the hypothesised object's identity is determined by the invariant descriptors measured, so that systems with relatively large model bases can be constructed (currently, of the order of thirty).

---

[1] Department of Computer Science, University of Iowa, Iowa City, Iowa.

[2] Oxford University Robotics Research Group, Parks Rd, Oxford

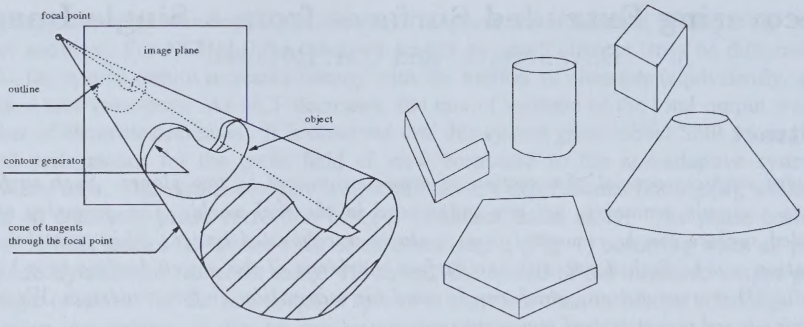


Figure 1: The left hand figure shows the outline and contour generator of a curved object, viewed from a perspective camera. The right hand figure shows a range of examples of extruded surfaces, most with vertex at infinity.

1.2 Recognising curved surfaces

The indexing systems described assume either that the model base contains only plane or polyhedral objects, or that depth data is available. Recognising curved surfaces in single images presents a particularly difficult problem, as the change in appearance of a curved surface as it is imaged from different viewing positions is not easily analysed. Throughout the paper, we assume an idealised pinhole camera. These cameras possess a *focal point* and an *image plane*. For each point in space, there is a line through that point and the focal point; the point in space appears in the image as the intersection of this line with the image plane - figure 1 illustrates such a camera. An orthographic view occurs when the pinhole is "at infinity".

If the focal point is fixed and the image plane is moved, the resulting distortion of the image is a collineation. In what follows, it is assumed that neither the position of the image plane with respect to the focal point nor the size and aspect ratio of the pixels on the camera plane is known, so that the image presented to the algorithm is within some arbitrary collineation of the "correct" image. In this model, the image plane makes no contribution to the geometry, and its position in space is ignored.

The *outline* of a surface is a plane curve in the image, which itself is the projection of a space curve, known as a *contour generator*. The contour generator is given by those points on the surface where the surface turns away from the image plane; formally, the ray through the focal point to the surface is tangent to the surface. As a result, at an outline point, if the relevant surface patch is visible, nearby pixels in the image will see vastly different points on the surface, and so outline points usually have sharp changes in image brightness associated with them. Figure 1 illustrates these concepts.

It is generally accepted that the problem of recovering surface geometry from an outline alone is intractable if the surface is constrained only to be smooth, or piecewise smooth, as in this case significant changes can be made to the surface geometry without affecting the outline from a given viewpoint. As a result, an important part of the problem involves constructing as large a class of surfaces as possible that can either be directly recognised or usefully constrained from their outline alone. There have been advances in a number of special cases:[5, 8, 2] describe a systems for recognising rotationally symmetric surfaces from outlines, [1, 8, 9, 14, 15] treat straight homogenous generalised cylinders and [6] describes recognising algebraic surfaces from outlines. All these systems emphasize surface geometry in recognition.

Other systems, such as those of [12, 10] emphasize cues such as color and surface lightness over geometry. These cues are particularly effective in dealing with objects whose geometry is ill-defined or hard to describe (such as shirts or jerseys). Such cues cannot, however, be completely divorced from shape, as by themselves they are not particularly distinctive. Furthermore, perspective effects make it difficult to frame colour features in a truly shape independent fashion; for example, foreshortening can have substantial effects on a colour histogram.

Ideally, a representation of a surface would contain information both about shape and about surface pattern features, such as colour and surface lightness. The colour information should be provided as a pattern on a representation of the surface (i.e. referred to its position on the surface in some canonical frame), so that foreshortening effects and the like can be discounted. This paper shows how to construct such a representation from a single image of an extruded surface.

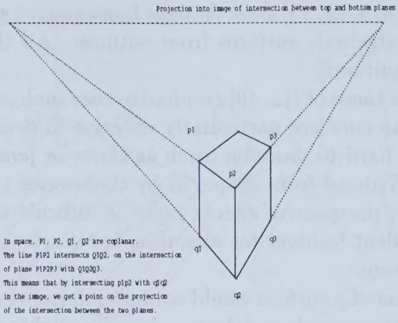
2 Recovering extruded surfaces

An *extruded surface* is a surface formed by a section cut from a general cone by two planes (see figure 1) in such a way that the section of surface does not include the vertex of the cone. This is the projective generalisation of the a surface formed by a system of parallel lines, with plane ends (such a surface can be extruded from a nozzle). Extruded surfaces, and surfaces made up from extruded components, are extremely common - examples include most tin cans, boxes, books, and many plastic bottles.

The outline of an extruded surface consists of a system of line segments (possibly empty for some views), the projection of one plane section, and the (possibly occluded) projection of the other plane section; in particular, for all focal points outside a simple "forbidden zone", which lies between the top and bottom sectioning planes and to the object's side of their line of intersection, one or the other plane section is completely visible, and there are usually at least two line segments in the outline.

The geometry of an extruded surface can be completely reconstructed in a single image (given the focal point is generic), because it is so simple. Consider one of the plane sections, taken with the line on that plane where the two sectioning planes intersect; call the plane chosen the *defining plane* and the configuration of plane section and line a *defining plane section* or D.P.S. The complete projective geometry of the surface can be represented by a defining plane sections, as it is possible can choose a projective transformation of space that fixes the D.P.S., and transforms the vertex to any given point off the defining plane section and the other sectioning plane, which must pass through the line in the D.P.S., to any given plane not the defining plane nor containing the vertex. As a result, the surface can be reconstructed by taking a D.P.S, choosing a second sectioning plane through the line on the D.P.S. arbitrarily, and choosing an arbitrary vertex in space. The section of surface that lies between the planes and does not include the vertex is the reconstruction.

As at least one plane section is visible in most images, it is possible to recover the surface's projective geometry the projection of the line of intersection between the planes in space can be recovered in the image. This can be done, if the projection of the vertex


Projection into image of intersection between top and bottom planes
p1
p3
p2
q1
q3
q2

In space, P1, P2, Q1, Q2 are coplanar.
The line P1P2 intersects Q1Q2, on the intersection
of plane P1P2P3 with Q1Q2Q3.
This means that by intersecting p1p2 with q1q2
in the image, we get a point on the projection
of the intersection between the two planes.

Figure 2: Determining the projection into the image of the line of intersection of the two sectioning planes.

is known, using the following approach (which is illustrated in figure 2):

- Mark three points, p_1, p_2 and p_3 on one plane section. These correspond to the three coplanar points P_1, P_2 and P_3 in space.

- Construct the corresponding points on the other plane section (for objects with a convex cross-section, self-occlusion is not, in fact, a problem here; see below). Call these points p'_1, p'_2, p'_3, and the corresponding coplanar points in space P'_1, P'_2, P'_3.

- Intersect the image lines p_1p_2, $p'_1p'_2$ to form q_{12} and the lines p_2p_3, $p'_2p'_3$ to form q_{23}. Since P_i, P'_i, P_j, P'_j must be coplanar (the lines $P_iP'_i$ and $P_jP'_j$ are rulings of the surface, and hence pass through the vertex), the intersections in the image are projections of actual intersections in space of the lines P_1P_2, $P'_1P'_2$, etc.

- Since P_1P_2 lies on the top plane, and $P'_1P'_2$ lies on the bottom plane, their intersection must lie on the intersection between the two planes. Thus, the projected line of intersection must pass through q_{12} and through q_{23}.

The aspect of this construction most likely to provide problems for early vision is the identification of corresponding points, which is complicated by self-occlusion. This problem is relatively easy to deal with if the cross-section of the cone is convex. For such cross-sections, which are well represented in applications, reasoning from a plane cross-section of the object is sufficient. It is apparent that only three possible cases of self-occlusion are significant:

1. one sectioning plane is occluded, and one is visible;
2. both sectioning planes are visible;
3. both are occluded.

Figure 3 shows how each case can occur. In case 3, it is possible to recover the projective geometry of the object only partially; as this case is "unusual" (from the position of the focal point), we concentrate on the other two cases.

For a convex object, top-bottom correspondences are also easily obtained. Consider a general plane through the vertex and through the focal point; such a plane (which

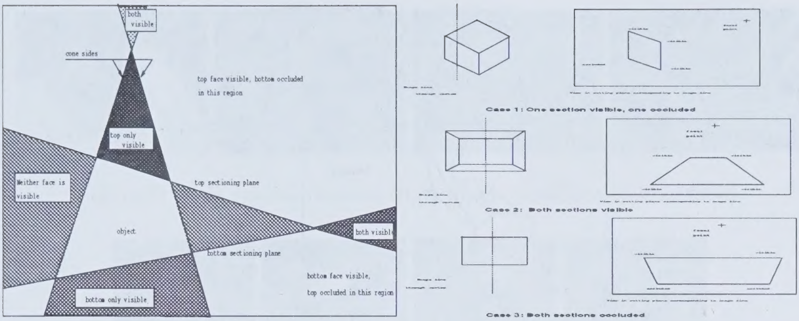

both visible
cone sides
top face visible, bottom occluded in this region
top only visible
Neither face is visible
top sectioning plane
object
both visible
bottom sectioning plane
bottom face visible, top occluded in this region
bottom only visible
Case 1: One section visible, one occluded
Case 2: Both sections visible
Case 3: Both sections occluded

Figure 3: The left hand figure shows how the possible cases of occlusion of both top and bottom section curves depend on the position of the focal point with respect to the cone, and with respect to the top and bottom sectioning planes; this figure shows the possible cases, for an object viewed in section. Reasoning from a section is sufficient if the cross-section is convex. The right hand figure shows the effects of occlusion on determining top-bottom correspondences, for a convex cross section and the three cases given in the text.

appears in the image as a line), and cuts the top and bottom sections in a number of points, some or all of which are visible in the image as points along a line. The cases are as follows:

- **One section visible:** if (say) the bottom section is occluded, all but one of the points of intersection between the given plane and the bottom section are occluded (figure 3). Hence, the number of visible intersections is odd, and points appear in a cyclic permutation of the order (u_1, u_2, v_2) (where u and v are either top or bottom respectively, and u_i and v_i correspond) along the image line.
- **Both sections visible:** if neither section is occluded, all points of intersection will be visible and there will be an even number of intersections between the outline and the image line, which appear in a cyclic permutation of the order (u_1, u_2, v_2, v_1) along the image line.
- **Both sections occluded:** if both sections are occluded, one point of intersection will be visible for both top and bottom sectioning plane, and there will be two intersections between the outline and the image line, which clearly appear in a cyclic permutation of the order (u_1, v_1) along the image line.

In practice, reconstruction proceeds as follows:

1. The cross-section of the cone is determined, by finding the image curve corresponding to the intersection of the cone and the top plane. This curve gives a plane curve within a projective transformation of the original section, and is projected into some arbitrarily chosen plane in space
2. The projection of the vertex is determined in the image, by intersecting line segments in the outline. An arbitrary vertex is chosen in space.

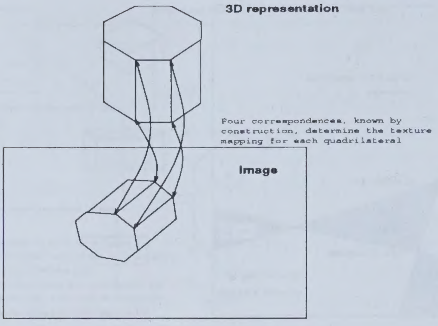

3D representation
Four correspondences, known by construction, determine the texture mapping for each quadrilateral
Image

Figure 4: Patterns and surface markings are transferred to the representation of the extruded surface, using the fact that four known correspondences determine a plane to plane projection. These correspondences are determined by construction.

3. The image line that is the projection of the line where the top and bottom planes intersect, is determined (figure 2, and above), and projected onto the same plane as the cross-section, ensuring that the configuration of line and outline is projected as one.
4. The cone is constructed by constructing lines through the vertex and points on the plane cross-section.
5. The second plane is chosen; any plane passing through the line of intersection, and not through the vertex, will do. The lines passing through points on the top cross-section and the vertex mark out the bottom cross-section.

2.1 Recovering surface markings

To transfer surface markings from the image to the representation of the surface, it is convenient to break up the cross-section in the image into a polygonal approximation (the polygons can be arbitrarily small), and then note that this leads to a representation of the surface as a system of plane quadrilaterals. Finally, the positions of the vertices in the image to which the 3D vertices project, are known. This determines the projective transformation from the image quadrilateral to the surface quadrilateral (four points and their images determine a projective transformation), and this transformation applied to pattern points maps them onto the surface representation (figure 4). Clearly, if this texture mapping process works for grey-level surface markings, it will work for color surface markings as well. A number of color sequences exist, although production expense dictates that the examples given show results only for the grey level case.

3 Experimental results and discussion

This scheme has been implemented for images of real scenes. We use a simple manual process to mark outline points in images; in particular, the operator marks the two lines that determine a vertex, points on the top curve, and corresponding points on the bottom

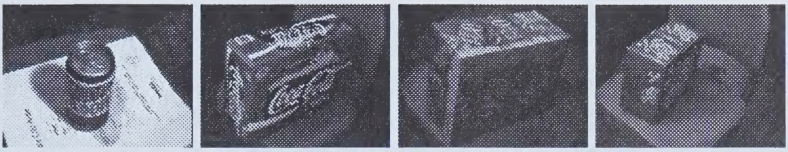


Figure 5: Four typical images, from which models were recovered.

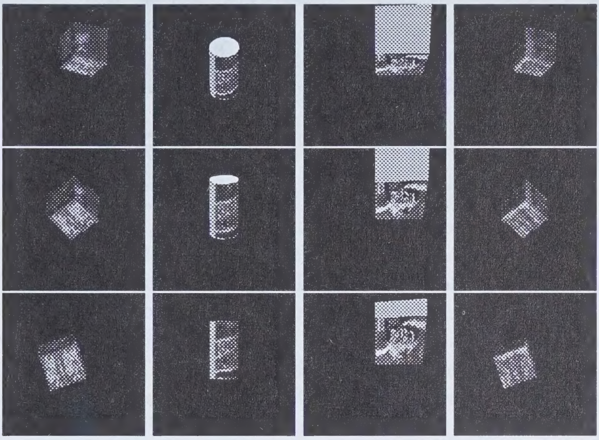


Figure 6: Frames from four motion sequences, created from the representations recovered using the techniques described in the text, showing boxes for soda cans tumbling, and a soda can tumbling. Note that the faces not visible in the image used for making models are rendered blank; constructing an integrated representation from a number of views of an object is an interesting question, at present open. Objects are scaled by hand, and rendered into a canonical frame. Note also the effects of image spatial quantization on one face of the box (which was severely foreshortened in the original image.

curve. This manual interface was used to allow a quick implementation to demonstrate the recovery process; we do not believe that it is essential. The representations extracted are 3D geometrical objects with associated surface markings, and therefore lend themselves well to display as a movie. Figure 5 shows typical images from which models are extracted; figure 6 shows frames from movies of tumbling objects. Note that the texture on the object is stable as the figure tumbles, indicating that the surface markings are being correctly extracted and placed; note further that, if the operator chooses a projective frame in which, for example, the soda cans have a circular cross section and roughly the right aspect ratio, the models are impressively realistic. Figure 7 shows two views of a simple 3D world constructed using a tool that places extruded models in space with respect to one another.

This form of representation can be used to support a hierarchical recognition system, that uses both shape and surface marking information to represent and recognise objects. As the geometry of objects of this form is so simple (as we have seen, it is completely determined by a plane curve and a line), objects can easily be indexed using a geometrical

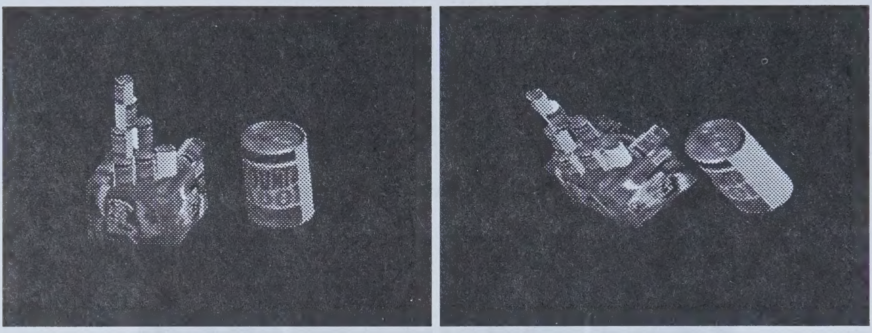


Figure 7: Two views of a three dimensional world created and rendered using the representations described in the text.

description. At that point, the surface marking information, which is in a canonical frame, can be used to generate further indexing information for the surfaces using the methods of, for example, Nayar and Bolle [10]. A judicious use of surface marking information is likely to break the projective ambiguity implicit in the geometrical reconstruction - for example, a container for cans of grape soda and a matchbox are projectively equivalent, but have different markings on their faces. Figures 8 show the canonical representations of face texture for three views of three different object faces. In general, these views look the same for different views of the same faces and different for views of different faces, and so should allow effective indexing. In fact, constructing an indexing process that works reliably for a large number of faces is not fully solved; much of the difficulty appears to stem from spatial quantisation noise, as when a face is strongly foreshortened, an image pixel may correspond to a large clump of pixels in the canonical representation. Constructing robust indexing techniques that use the surface marking information in these representations is the subject of active research.

Acknowledgements

Supported in part by the National Science Foundation under award no. IRI-9209729, in part by a grant from United States Air Force Office of Scientific Research AFOSR-91-0361, and by a National Science Foundation Young Investigator Award with matching funds from GE, Rockwell International and Tektronix. This idea has origins in both authors' collaborative work with Joe Mundy and Andrew Zisserman. DAF was encouraged to consider surface markings by a discussion with Shree Nayar. Thanks to David Mumford for helpful and informative conversations. Rodney André built the tool for modeling worlds of extruded surfaces.

References

[1] Binford, T.O., Levitt, T.S., and Mann, W.B., "Bayesian inference in model-based machine vision," in Kanal, L.N., Levitt, T.S., and Lemmer, J.F., *Uncertainty in AI 3*, Elsevier, 1989.

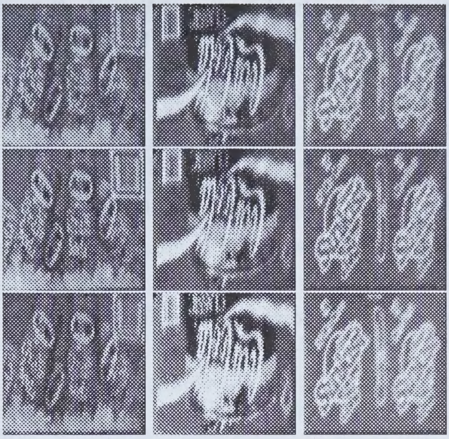


Figure 8: Canonical representations of face markings for three views of three different object faces; note that different views of the same face look the same, and views of different faces look different.

[2] Dhome, M., LaPreste, J.T, Rives, G., and Richetin, M. "Spatial localisation of modelled objects in monocular perspective vision," *Proc. First European Conference on Computer Vision*, 1990.

[3] Forsyth, D.A., Mundy, J.L., Zisserman, A.P., Coelho, C., Heller, A. and Rothwell, C.A. "Invariant Descriptors for 3-D Object Recognition and Pose," *PAMI*-13, No. 10, p.971-991, October 1991.

[4] Forsyth, D.A., Mundy, J.L., Zisserman, A.P. and Rothwell, C.A. "Applications of invariant theory in vision," In Kapur, D. and Donald, B.R., (eds) *Proceedings Workshop on Integration of Symbolic and Numerical Methods, Saratoga N.Y.*, Academic Press, 1992.

[5] Forsyth, D.A., Mundy, J.L., Zisserman, A.P. and Rothwell, C.A. "Recognising Curved Surfaces from their Outlines," Proceedings ECCV2, p.639-648, 1992.

[6] Forsyth, D.A., "Recognizing Algebraic Surfaces from their Outlines," *Accepted for Publication, International J. of Computer Vision*, 1993.

[7] Lamdan, Y., Schwartz, J.T. and Wolfson, H.J. "Object Recognition by Affine Invariant Matching," Proceedings CVPR88, p.335-344, 1988.

[8] Liu J., Mundy J.L., Forsyth D.A., Zisserman A. and Rothwell C.A., "Efficient Recognition of Rotationally Symmetric Surfaces and Straight Homogeneous Generalized Cylinders", *CVPR*, 1993.

[9] Ponce, J. "Invariant properties of straight homogenous generalised cylinders," *IEEE Trans. Patt. Anal. Mach. Intelligence*, **11**, 9, 951-965, 1989.

[10] Nayar, S.K. and Bolle, R. "Reflectance Ratio: A Photometric Invariant for Object Recognition," *Proc ICCV-4*, Berlin, 1993.

[11] Rothwell, C.A., Zisserman, A., Mundy, J.L. and Forsyth, D.A. "Efficient Model Library Access by Projectively Invariant Indexing Functions", Proceedings CVPR92, p.109-114, 1992.

[12] Swain, M.J. and Ballard, D.H., "Color Indexing," *International Journal of Computer Vision*, **7**, 1, 11-32, 1991.

[13] Taubin, G. and Cooper, D.B. "Recognition and Positioning of 3D Piecewise Algebraic," Proceeding DARPA Image Understanding Workshop, p.508-514, September 1990.

[14] Ulupinar, F, and Nevatia, R. "Shape from Contour using SHGCs," *Proc. ICCV*, Osaka, 1990.

[15] Ulupinar, F, and Nevatia, R. "Recovering shape from contour for constant cross-section generalisd cylinders," *Proc. CVPR*, Mauii, 1991.

[16] Weiss, I. "Projective Invariants of Shapes," Proceedings DARPA Image Understanding Workshop, p.1125-1134, April 1988.

Force-Guided Assembly

Michael Peshkin[1]
Mechanical Engineering
Northwestern University
Evanston IL 60208
USA

Ambarish Goswami[2]
INRIA-RA (Bat. Hitella)
46 Avenue Felix Viallet
38031 Grenoble Cedex 1
France

Joe Schimmels[3]
Mechanical Engineering
Marquette University
Milwaukee WI 53233
USA

Abstract -

Two major trends have emerged in the field that deals with the control of robotic manipulators in interactive tasks. They are *fine motion planning* and *force-guided assembly*. Somewhat computational in spirit, *fine motion planning* assumes a simple force control law for the manipulator and computes a complicated trajectory that is robust against control and sensor uncertainties, and errors in actuation, robot model, and the environment model.

In *force-guided assembly,* we instead assume a simple nominal motion which would result in a successful parts mating in the absence of any positional error. The next step is to invent a control law (perhaps given by an accommodation matrix) that would map the forces resulting from various errors into error-corrective velocities.

It is instructive to compare force-guided assembly with other major assembly strategies, and to show where it fits into the general scheme of robot control. To that end, we briefly trace the development of fine motion planning and of force-guided assembly. We describe impedance/admittance control, and the several special forms of impedance/admittance control which are in use. Finally, we discuss accommodation control as a special form of impedance/admittance control most applicable to force-guided assembly.

History of Fine Motion Planning

We begin with a brief outline of the research in fine motion planning. An exhaustive study of fine motion planning is beyond our scope and interested readers may refer to [4, 6, 7, 12, 13].

The phrase "fine motion planning" was originally used to contrast it from gross motion planning, an example of latter being the planning of robot trajectory through free space. Two different developments complicated this distinction. First, the problem of fine motion planning was generalized to include the problem of robots making gross motions through a space cluttered with obstacles (as described below), thereby making the distinction between fine motion planning and gross motion planning less rigid. Second, perhaps accidentally, the phrase fine motion planning came to be known as a particular type of fine motion planning where the goal was to assume a simple control law and then compute a complicated motion trajectory. The other strategy (that of assuming a simple trajectory and computing an appropriate control law) generically came to be known as force control.

---

[1] peshkin@nwu.edu
[2] goswami@imag.fr
[3] jms@jms.meie.mu.edu

Fine motion planning was originally concerned with the automatic generation of motion strategies for mechanical assembly. The unsuitability of pure position-controlled robots in assembly was recognized early in the motion planning research [7]. Researchers realized that position-controlled robots are unable to either avoid or overcome unexpected collisions. Perhaps more important, these robots cannot recognize the success (i.e., arrival at the goal position) of a task. Use of force/torque data during a task vastly enlarges the range of tasks that may be addressed by fine motion planning.

Inoue [10] suggested the deliberate introduction of a large systematic error in the assembly such that the smaller random uncertainties are absorbed and the relative positions of the mating parts become much more predictable. As an example, in a peg-in-hole operation one may introduce a relatively large leftward translation and a tilt to the peg such that it always hits the left side of the hole and is inclined from the vertical. The peg can then slide along the surface until it encounters the hole. A backward tilt will then make the peg vertical and securely place it inside the hole.

Lozano-Perez [7] and Taylor [28] suggested a *skeleton search strategy* that can relate error estimates to a particular strategy specification. In other words, depending on the error estimate, the planner would choose between, for instance, a *compliant motion plan* and a *guarded move*. A compliant motion is one that enables the robot to slide along a solid surface, a good example being the task of drawing a line on a chalkboard. Guarded motions are executed to detect a collision with an object (through a force sensor). The task of gently placing a book on a table can benefit from a guarded move.

Motion planning research has generalized the interactive environment encountered in a typical assembly task to that of an imperfectly controlled mobile robot in an environment cluttered with obstacles. The objective is to compute a continuous collision-free trajectory of the robot from the initial position to the goal position subject to imperfect sensing by the robot. The uncertainties in modeling the environment and the robot have not been seriously considered until more recently [4] [23], partly because of the complexity of the problem.

Udupa [29] first suggested the advantage of transforming the mobile robot problem to the problem of steering a point robot among obstacles which have been enlarged to account for the shrinking of the robot. This idea was formalized and extended to higher dimensions (in order to incorporate all six degrees-of-freedom of motion) by Lozano-Perez [11]. The artificial environment cluttered with enlarged obstacles and a moving point robot was named the *configuration space* of the robot.

Lozano-Perez, Mason, and Taylor [12] then suggested the useful idea of *backchaining*. Given a clearly stated goal condition, the backchaining method constructs a region in the configuration-space of the robot from where, in the presence of sensing and control uncertainty, it is always possible to reach the goal position. The region in configuration space is called the *pre-image* of the goal position, for the given errors. This approach to fine motion planning is often called the LMT formulation after the authors. Erdmann [7] developed an algorithm for automatically computing the preimage for an assembly. For complicated tasks, a single-step motion may not be sufficient for the assembly. In that case, one may progressively backchain from the goal position until the starting position is reached. The starting position of each segment of the chain is the goal position of the next segment. There is, however, no effective procedure to systematically compute the motion commands. Also, it is not always easy to recognize that a goal position has been reached. Erdmann's work addresses this latter issue of goal recognizability and also extends the theory of configuration space to include frictional contacts between the mating parts [7].

Fine motion planning, as described above, is a very conservative approach to motion planning since it requires *guaranteed* success of the planner. In situations where success cannot be guaranteed, the traditional fine motion planner fails to generate a motion strategy. There are situations where it is impossible to predict the success of a task beforehand. A different approach, called an *error detection and recovery* (EDR) plan has been found useful in these situations [4]. EDR plans work in those environments where a successful assembly is often possible but may not be guaranteed.

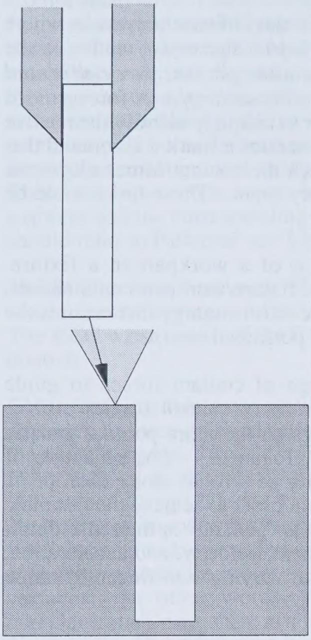


Figure 1: An assembly situation where conventional Fine Motion Planning Fails and Error Detection and Recovery (EDR) strategy might succeed. The motion of the peg is such that the resulting force is within the friction cone of the contacting surface and the peg "sticks". EDR strategy tries to recover from the situation by trying to move toward the left or right [5].

As an example ([4]) in the assembly operation shown in Figure 1, the peg's velocity might be inside the friction cone of the mating surface. This will result in sticking of the peg[4]. A traditional fine motion planner would try to compute trajectories that are guaranteed to lie outside the friction cone . If the velocity uncertainty of the peg is relatively large, the planner would fail to find a trajectory. An EDR plan, in this situation, would plan for possible occurrences of sticking and would try to move to the left or to the right to help the peg recover from its stalled state.

A somewhat different situation occurs when, due to manufacturing tolerances there exists a possibility for the peg to be larger than the whole. If a traditional motion planner wants to generate a guaranteed trajectory, it fails to do so. This is wasteful, since in reality, only in a small percentage of cases, the peg will actually become larger than the hole and the assembly will be physically impossible. In cases such as this, the goal of the EDR planner is to try to accomplish the task or fail recognizably. The EDR planner also accepts an extended configuration space into which parameter variation of the mating parts can be incorporated [5].

Even with its sophisticated theoretical backbone, motion planning is still a difficult task and is computationally intensive. The so called findpath algorithm is complex. The fastest available (but not implemented) algorithm for the find-path problem is due to Canny [3, 5].

---

[4] Since fine motion planners typically uses a diagonal accommodation matrix (possessing a center of accommodation at the tip of the peg), the forces and velocities are in the same direction. Non-diagonal accommodation matrices result in non-intuitive force-motion relationships. In particular the cone of *velocities* which result in sticking will not coincide with the friction cone, which is the cone of *forces* that result in sticking.

History of force-guided assembly

The term *force-guided assembly* was introduced to refer to the class of assembly tasks where the force/torque information is used for guiding the workparts to a successful mating in the presence of positional errors [21, 24, 25, 26]. (A very similar phrase, *force-directed insertions*, was used by Strip [83] to refer to a different assembly strategy.) A force-guided assembly strategy exploits the force/torque information in order to uniquely identify the relative positional errors between the workparts, and subsequently generates a motion command that reduces the error. There exists a large class of tasks in which the contact forces alone can guide the robot to the goal position without any other sensory input. These tasks could be accomplished by a suitably formulated force control strategy.

One example of such a task is the robot-assisted insertion of a workpart in a fixture. Schimmels and Peshkin [25] showed that for a large variety of fixture/workpart combinations (called deterministic fixtures), it is possible to devise a force control strategy that ensures the unique positioning of a workpart in a fixture in the presence of positional errors.

Force-guided assembly takes explicit or implicit advantage of contact forces to guide mismatched workparts to successful mating. The relevant force control framework for *impedance/admittance control* (which is often improperly given the more popular generic name of "force control") and was originally proposed by Hogan [9]. The objective of impedance control is to control the general dynamic behavior of a robot along each of its degrees-of-freedom. A suitable impedance function is chosen in order to achieve the desirable dynamic behavior of the manipulator. The dynamic behavior is "general" in the sense that it may relate forces generated by a robot to the input displacement, velocity, and acceleration. Moreover, the impedance control scheme may include time-varying non-linear dynamic behavior of the robot.

The philosophy of impedance control is based on physical systems theory. It starts by pointing out the fact that along any degree-of-freedom one may control the force or the motion (displacement, velocity, and acceleration) of a manipulator, but not both.

Interaction of a robot with its environment is indicated by a transfer of power between them. Power is computed as the product of two conjugate power variables – an *effort* variable (force) and a *flow* variable (velocity). Impedance-type physical systems accept a flow variable as input and produce an effort variable. Admittance-type systems accept an effort variable and produce a flow variable. The distinction between the impedance and the admittance characterizations of a system is one of causality and is not always apparent for simple linear systems. Causality is, however, a fundamental characteristic of non-linear systems since a non-linear system is often not describable both as an impedance and an admittance. Two interacting systems must complement each other in terms of causality. Hogan argued that manipulators interact with objects of mainly admittance type. Therefore they must be modeled as impedances.

A position-controlled robot emulates, at its endpoint, an infinite impedance (zero admittance) and therefore cannot be deflected from its path as long as the deflecting force is small compared to the maximum reaction force that the motors can generate. A force-controlled robot is "soft" at its endpoint because it emulates a zero impedance (infinite admittance). Contact with objects will readily cause it to yield. Stiffness/compliance control and damping/accommodation control are also special cases of impedance control. Both of these schemes represent different types of force control and differ only in the way a robot yields while interacting with the environment. A robot under stiffness/compliance control emulates the behavior of a generalized spring whereas a robot under damping/accommodation control emulates the behavior of a generalized damper.

A widely accepted strategy in compliance control for the error-corrective assembly of peg-in-chamfered-hole is to select an compliance matrix possessing a "center" at the tip of the grasped peg [15, 16, 27, 31, 32]. A mechanical device, which is called an RCC (Remote Center of Compliance) wrist is used to implement this control strategy. RCC wrists have been successful in the prototypic assembly task of peg-in-hole and has, consequently, been analyzed in great detail. The *center of compliance* is the point in the workspace of a robot where the compliance matrix is diagonal. A diagonal compliance matrix generally means that, at the tip of the peg, the translational and rotational components of the displacements are decoupled. In other words, a force resulting from a translational mismatch between the peg and the hole produces only a translational error-correcting displacement. Similarly a torque resulting from an angular misalignment will generate only an angular displacement in the same direction. There are some subtle issues involved in the structures of accommodation and compliance matrices and the corresponding force-motion characteristics of the system. Interested readers should refer to Patterson and Lipkin [17, 18].

The RCC wrist is interesting to us for another reason. A typical design of an RCC wrist makes use of passive springs and this makes it fast and stable (as we will discuss in Section 1.5). The RCC wrist is one of the first applications of a physically passive device for robot force control.

Before leaving this section we will describe an interesting (although not very practical) approach to force-guided motion strategy – the strategy of hybrid position/force control. As the name suggests, a hybrid control strategy is capable of controlling robot's position or force along each of its degrees-of-freedom. The concept of hybrid control is due to Inoue [10], Raibert and Craig [22], and Mason [13] [14]. The control scheme recognizes the fact that the environment imposes positional and force constraints on a robot along different degrees of freedom in its task-space. As an example, for a workpart sliding along a solid surface, there is a position constraint along the mutual normal at the contact point of the two contacting surfaces. In other words, the workpart cannot move into the solid surface. In a complementary way, there is a force constraint along the tangential direction, implying that the force in that direction will be always zero. These constraints, which are dictated by the environment are called the *natural constraints*. The goal of hybrid control is to endow force control to the robot along the normal direction (i.e., the direction of the position constraint) and position control along the tangential direction (i.e., the direction of the force constraint). One may wish the robot to follow a particular force control law or a particular position control law. These control laws represent what Mason called the *artificial constraints* of the robot [13].

One of the limitations of hybrid control is that it may not always be possible to decompose the task-space into mutually orthogonal directions of position constraint and force constraint. As an example, for a workpart in contact with multiple fixture elements, there are no clear-cut separate directions for force control and position control. Even for simpler tasks, positional uncertainty makes it impossible to pre-specify distinct force control and position control directions, thereby preventing practical use of hybrid control.

Accommodation Control for Force-Guided Assembly

Accommodation control is a special case of impedance/admittance control where the robot only needs to display the characteristics of generalized dampers. The desired accommodation characteristics along the robot's task-space degrees-of-freedom may be specified with an accommodation matrix or an accommodation function (if non-linear). A typical block diagram for the implementation of accommodation control is shown in Figure 2.

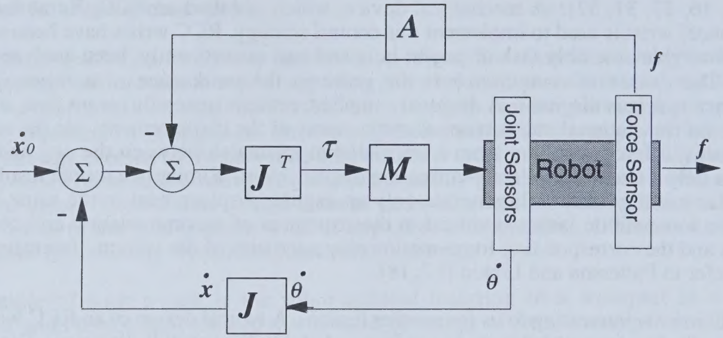


Figure 2: A control architecture for implementing accommodation control.

Let us now introduce the particular form of linear control law that we wish to implement for error-corrective force-guided assembly tasks [19, 20, 21]. Suppose a workpart is held by the robot (refer to the place-in-detent operation described in Figure 3), and is moving by virtue of the gross motion of the robot, with a nominal velocity v_0 in the absence of any assembly force. v_0 is the velocity of the robot/workpart under pure position control. When the workpart comes in contact with the other (perhaps misaligned) mating part, its resultant velocity v may be expressed as

$$v = v_0 + Af \tag{1.1}$$

where f is the force on the workpart due to contact. A is the *accommodation matrix* that the robot exhibits to the environment.

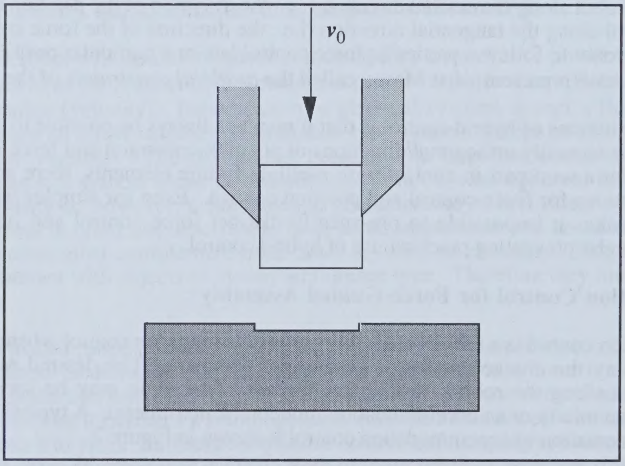


Figure 3: *Place-in-Detent* - An example of mechanical assembly. The square block (grasped by a robot) has to be placed in the shallow detent of the workpart at the bottom. In an ideal situation with a perfect robot, perfect sensing and control, and perfectly manufactured workparts this assembly can be performed without difficulty. However in reality, it is not a trivial task to automate this assembly.

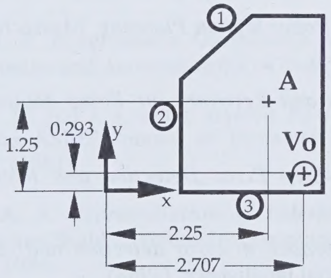
Af is a "velocity correction" which is superposed on the nominal velocity to generate the error-corrective velocity *v*. The force due to erroneous contacts contains information about what went wrong with the assembly. The main idea here is to exploit this information in order to rectify the positional mismatch of the assembly. The success of this control strategy lies, of course, in the proper choice of the accommodation matrix *A*.

The accommodation matrix suitable for a particular task is closely related to the task geometry. A matrix useful for a certain task may not be useful (in fact, it could be detrimental) for another. Matrices with a "center" are not necessarily useful for tasks other than a peg-in-hole operation. Therefore one should consider a larger class of matrices in order to identify the best suitable matrix for a given task. Schimmels and Peshkin [25] formulated a systematic means of identifying the bounds of force-guided assembly and a systematic approach to the design of a manipulator's accommodation matrix that guarantees force-guided assembly. They often find non-diagonal matrices to be suitable for assembly tasks, as in Figure 4 [25] [24].

The basic idea in accommodation control is to "map" an input force (due to erroneous contacts) to an error-correcting velocity. The mapping need not be linear, in which case, we may modify the equation as $v = v_0 + A(f)$, where $A(f)$ is a non-linear accommodation function of f. Asada [2] suggested that one should be able to train a neural network to implement the precise non-linear accommodation function, if enough force-velocity pairs are available for the purpose of training the network. Gullapilli et al. [8] implemented a neural network based accommodation control scheme on a robot and obtained encouraging results.

In order to compute an explicit accommodation matrix, to build a look-up table, or for the purpose of training a neural network, we need to generate a list of input forces and corresponding successful velocities. This is a crucial step in the success of the control scheme and it involves various subtleties.

Ahn et al. [1] suggested a learning scheme for the generation of a large set of force-velocity pairs. In their approach, an actual experimental set-up is used where force-velocity data for each successful operation is stored in a computer in real time. When a robot encounters an unknown force, it first tries to look up the force nearest to it in the stored list. In the absence of a reasonably close force data, it makes a move in a random direction and keeps track of the outcome.



$$\mathbf{A} = \begin{bmatrix} 7 & -11 & 5 \\ -11 & 21 & -9 \\ 5 & -9 & 4 \end{bmatrix}$$

Figure 4: Example of an accommodation matrix which makes it possible to perform force-guided assembly of a workpiece into a three-fixel fixture. The necessary accommodation matrix (referred to the origin of the coordinate system at the lower left) is **A** as given. With this accommodation matrix, no matter which fixel the workpiece first contacts, the consequent motion of the workpiece will serve to bring it into contact with the remaining fixels. (From [25])

Peshkin generated force-velocity pairs through what he called the *erroneous contact configurations* of a task [21]. This is a simple, intuitive, but a non-systematic way of generating a discrete representative set of possible erroneous situations of the workparts. The forces and error-correcting velocities corresponding to the configurations are generated from a geometric model by inspection. If an actual encountered force does not fall into the category of one of the configurations considered, there is no guarantee (even theoretically) that the assembly will succeed. In simple tasks, however, most realistic forces can found as linear combinations of the forces from the erroneous contact combinations.

Vougioukas and Gottschlich [30] suggested a more systematic way of generating force-velocity pairs from geometric models of the workparts. Their approach, based on the configuration space (described in the next section in relation with fine motion planning) of the mating workparts, accounts for workpart model errors and velocity uncertainties. They use a Monte-Carlo method to randomly select a representative set of forces from the "space" of possible erroneous force/torques. Corresponding error-correcting velocities are then computed from the geometric models. Here again, the determination of the "closeness" of a sensed force/torque to one already generated is a subtle issue and is currently being investigated [personal communication with Gottschlich].

Acknowledgments

The research of the authors mentioned in this paper was performed with the support of the National Science Foundation, grant DMC-8857854. Much of the text of the historical survey is drawn from the PhD dissertation of Ambarish Goswami, Northwestern University, 1993.

Bibliography

1. Ahn, D. S., H. S. Cho, K. Ide, F. Miyazaki and S. Arimoto. *Strategy Generation and Skill Acquisition for Automated Robotic Assembly Task.* IEEE International Symposium on Intelligent Control. Arlington, VA IEEE Press (1991)

2. Asada, H. *Teaching and Learning of Compliance Using Neural Nets: Representation and Generation of Nonlinear Compliance.* Proceedings, IEEE International Conference on Robotics and Automation. Cincinnati, OH (1237-1244) IEEE Press (1990)

3. Canny, J. F. *The Complexity of Robot Motion Planning.* Massachusetts Institute of Technology. Cambridge, MA (1988)

4. Donald, B. R. *Error Detection and Recovery for Robot Motion Planning with Uncertainty.* Ph. D., MIT (1987)

5. Donald, B. R. *Planning Multi-Step Error Detection and Recovery Strategies.* (1987)

6. Donald, B. R. *A geometric approach to error detection and recovery for robot motion planning with uncertainty.* Artificial Intelligence (1988)

7. Erdmann, M. A. *On motion planning with uncertainty.* MS Thesis, Massachusetts Institute of Technology (1984)

8. Gullapilli, V., R. A. Grupen and A. G. Barto. *Learning Reactive Admittance Control.* IEEE International Conference on Robotics and Automation. 1992 (pp1475-1480) IEEE Computer Society Press (1993)

9. Hogan, N. *Impedance Control: An Approach to Manipulation.* ASME Journal of Dynamic Systems, Measurement, and Control **107**:1-24 (1985)

10. Inoue, H. *Force Feedback in Precise Assembly Tasks.* Artificial Intelligence: An MIT Perspecive. Brown ed. MIT Press. (1979)

11. Lozano-Perez, T. *Motion Planning for Simple Robot Manipulators.* Third International Symposium on Robotics Research. Paris MIT Press (1985)

12. Lozano-Perez, T., M. T. Mason and R. H. Taylor. *Automatic synthesis of fine-motion strategies for robots.* International Journal of Robotics Research **3**(1):3-24 (1984)

13. Mason, M. M. *Compliance and Force Control for Computer Controlled Manipulators.* IEEE Transactions on Systems, Man, and Cybernetics **SMC-11**(6) (1981)

14. Mason, M. M. *Compliant Motion.* Robot Motion: Planning and Control. Brady ed. The MIT Press. Cambridge, Massachusetts (1982)

15. Nevins, J. L. and D. E. Whitney. *The Force Vector Assembler Concept.* First Internation Conference on Robots and Manipulator Systems. Udine, Italy (1973)

16. Nevins, J. L. and D. E. Whitney. *Computer Controlled Assembly.* Scientific American. **238**: (1978)

17. Patterson, T. and H. Lipkin. *A Classification of Robot Compliance.* ASME Design Technical Conferences - 21st Biennial Mechanisms Conference. Chicago, IL (pp315-322) ASME (1990)

18. Patterson, T. and H. Lipkin. *Structure of Robot Compliance.* ASME Design Technical Conferences - 21st Biennial Mechanisms Conference. Chicago, IL (pp307-314) ASME (1990)

19. Peshkin, M. A. *Programmed compliance for error correction during robotic assembly.* 3rd IEEE 1988 International Symposium on Intelligent Control. Arlington, VA (1988)

20. Peshkin, M. A. *Programming a Damping Matrix for Error-Corrective Assembly.* Proceedings of the 4th International Symposium on Intelligent Control. Albany, NY (1989)

21. Peshkin, M. A. *Programmed Compliance for Error-Corrective Assembly.* IEEE Transactions on Robotics and Automation **6**(4):473-482 (1990)

22. Raibert, M. H. and J. J. Craig. *Hybrid Position/Force Control of Manipulators.* Transactions of the ASME Journal of Dynamic Systems, Measurement, and Control **102**(June):126-133 (1981)

23. Requicha, A. A. *Representation of Tolerances in Solid Modeling: Issues and Alternative Approaches.* Solid Modeling by Computer: From Theory to Application. Plenum Press. New York (1984)

24. Schimmels, J. M. and M. A. Peshkin. *Synthesis and Validation of Non-Diagonal Accommodation Matrices for Error-Corrective Assembly.* Proc. IEEE International Conference on Robotics and Automation. Cincinnati IEEE Press (1990)

25. Schimmels, J. M. and M. A. Peshkin. *Admittance Matrix Design for Force Guided Assembly.* IEEE Transactions on Robotics and Automation **Vol. 8**(No. 2):213-227 (1992)

26. Schimmels, J. M. and M. A. Peshkin. *The Robustness of an Admittance Control Law Designed for Force-Guided Assembly to the Disturbance of Contact Friction.* IEEE Intl. Conf. on Robotics and Automation. Nice, France (2361-2366) IEEE Press (1992)

27. Simunovic, S. *Force Information in Assembly Processes.* Fifth International Symposium on Industrial Robots. Chicago Society of Manufacturing Engineers (1975)

28. Taylor, R. H. *The Synthesis of Manipulator Control Programs from Task-Level Specifications.* (1976)

29. Udupa, S. M. *Collision Detection and Avoidance in Computer Controlled Manipulators.* Ph. D. Thesis, California Institute of Technology (1977)

30. Vougioukas, S. and S. Gottschlich. *Automatic Synthesis and Verification of Compliance Mappings.* IEEE International Conference on Robotics and Automation. Atlanta, GA (pp491-496) IEEE Computer Society Press (1993)

31. Watson, P. C. *A multidimensional system analysis of the assembly process as performed by a manipulator.* First North American Robot Conference. Chicago (1976)

32. Whitney, D. E. *Quasi-Static Assembly of Compliantly Supported Rigid Parts.* ASME J. Dynamic Systems, Measurement, and Control. **104**:65-77 (1982)

Derivation of Depth Maps Through Controlled Active Exploration

Christopher E. Smith and Nikolaos P. Papanikolopoulos

Artificial Intelligence, Robotics, and Vision Laboratory
Department of Computer Science
University of Minnesota
Minneapolis, MN 55455

ABSTRACT

Accurate knowledge of depth continues to be of critical importance in robotic systems. Without accurate depth knowledge, tasks such as inspection, tracking, grasping, and collision-free motion planning prove to be difficult and often unattainable. Of the various sensors that can be utilized to determine depth, visual sensors potentially provide information that is richer and more robust than other range sensors. Traditional visual depth recovery has relied upon techniques that require the solution of the correspondence problem or require known lighting conditions and Lambertian surfaces. In this paper, we present a technique for the derivation of depth from feature points on a target's surface using the Controlled Active Vision framework. We use a single visual sensor mounted on the end-effector of a robotic manipulator to automatically select feature points and to derive an estimate of the depth of those features using adaptive control techniques. Movements of the manipulator produce displacements that are measured using a Sum-of-Squared Difference (SSD) optical flow. The measured displacements are fed to the controller to alter the path of the manipulator and to refine the depth estimate.

1 Introduction

The computation of the relative and absolute depths of objects' feature points is essential information for the accurate execution of several robotic manipulation and inspection tasks. Traditional approaches to the problem of depth recovery have assumed that extremely accurate measurements of the camera parameters and the camera system geometry are provided *a priori*. Such information is not always available or, when it is available, not always accurate. Inaccuracies are introduced by positioning, path constraints, and environmental changes in the robotic system. Furthermore, camera calibration and the determination of camera parameters can be computationally expensive, time consuming, and error prone. In particular, many depth derivation techniques rely upon stereo vision systems that require careful geometry measurements and the solution of the correspondence problem, making the computational overhead prohibitive for real-time systems. Finally, many structure from motion algorithms use simple accidental motion of the camera that does not guarantee the best possible identifiability of the depth.

One solution to these problems can be found under the Controlled Active Vision framework. Instead of an accidental motion of the eye-in-hand system [8] [13] [14], we propose a controlled exploratory motion that provides identifiability of the depth parameter. An adaptive controller can be utilized to reduce the influence of such environmental, camera, and manipulator specific inaccuracies to provide accurate and reliable information regarding the depth of an object's feature points. This information may then be used to guide operations such as tracking, inspection, and manipulation [3] [12].

In this paper, we present a technique for the derivation of feature point depth based upon a Sum-of-Squared Differences (SSD) optical flow [1] under an adaptive control framework. The premise behind this technique is that, given an eye-in-hand robotic system, features are automatically selected, feature dependent trajectories are produced, and displacements of feature points are measured using the SSD optical flow. These displacements are compared to the predicted displacements that are derived using the current depth estimate. The errors from these comparisons are then used, in conjunction with the previous displacements, to update the depth estimate and to produce the next control input. The control input is derived such that the amount of the error removed in the next iteration will be maximized given environmental and manipulator specific constraints.

In this paper, we formulate the equations for optical measurements and the feature point selection scheme. Then, we derive the control and measurement equations used for the adaptive controller and elaborate on the selection of the features' trajectories. Finally, we discuss results from simulation and from experiments conducted using the Minnesota Robotic Visual Tracker (MRVT).

2 Vision-based Measurements and Feature Selection

2.1 Camera Model and Optical Flow

We assume a pinhole camera model with a world frame, R_W, centered on the optical axis (see Figure 1). We also assume a focal length f and a perspective projection. A point $\mathbf{P} = (X_W, Y_W, Z_W)^T$ in R_W, projects to a point $\mathbf{p}$ in the image plane with coordinates (x, y). We can define two scale factors s_x and s_y to account for camera sampling and pixel size, and include the center of the image coordinate system (c_x, c_y) given in frame F_A [12]. This results in the following equations for the actual image coordinates (x_A, y_A):

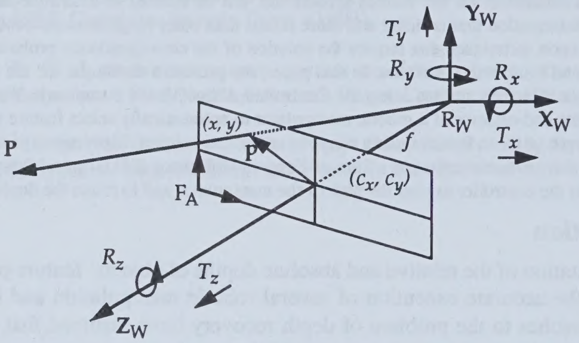


Figure 1: Perspective projection

$$x_A = \frac{fX_W}{s_x Z_W} + c_x = x + c_x \quad \text{and} \quad y_A = \frac{fY_W}{s_y Z_W} + c_y = y + c_y. \tag{2.1}$$

Any displacement of the point **P** can be described by a rotation about an axis through the origin and a translation. If this rotation is small, then it can be described as three independent rotations about the three axes $\mathbf{X}_W$, $\mathbf{Y}_W$, and $\mathbf{Z}_W$ [2]. We will assume that the camera moves in a static environment with a translational velocity (T_x, T_y, T_z) and a rotational velocity (R_x, R_y, R_z). The velocity of point **P** with respect to R_W can be expressed as:

$$\frac{d\mathbf{P}}{dt} = -\mathbf{T} - \mathbf{R} \times \mathbf{P}. \tag{2.2}$$

By taking the time derivatives and using equations (2.1) and (2.2), we obtain:

$$u = \left[\frac{x}{Z_W}T_z - \frac{f}{Z_W s_x}T_x\right] + \left[\frac{xys_y}{f}R_x - \left(\frac{f}{s_x} + \frac{x^2 s_x}{f}\right)R_y + \frac{ys_y}{s_x}R_z\right] \tag{2.3}$$

$$v = \left[\frac{y}{Z_W}T_z - \frac{f}{Z_W s_y}T_y\right] + \left[\left(\frac{f}{s_y} + \frac{y^2 s_y}{f}\right)R_x - \frac{xys_x}{f}R_y - \frac{xs_x}{s_y}R_z\right]. \tag{2.4}$$

We use a matching based technique known as the Sum-of-Squared Differences (SSD) optical flow [1]. For the point $\mathbf{p}(k-1) = (x(k-1), y(k-1))^T$ in the image $(k-1)$ where k denotes the kth image in a sequences of images, we want to find the point $\mathbf{p}(k) = (x(k-1)+u, y(k-1)+v)^T$. This point $\mathbf{p}(k)$ is the new position of the projection of the feature point $\mathbf{P}$ in image k. We assume that the intensity values in the neighborhood N of $\mathbf{p}$ remain relatively constant over the sequence k. We also assume that for a given k, $\mathbf{p}(k)$ can be found in an area Ω about $\mathbf{p}(k-1)$ and that the velocities are normalized by time T to get the displacements. Thus, for the point $\mathbf{p}(k-1)$, the SSD algorithm selects the displacement $\Delta \boldsymbol{x} = (u, v)^T$ that minimizes the SSD measure

$$e(\mathbf{p}(k-1), \Delta x) = \sum_{m, n \in N} \begin{matrix} [I_{k-1}(x(k-1)+m), y(k-1)+n) - \\ I_k(x(k-1)+m+u, y(k-1)+n+v)]^2 \end{matrix} \qquad (2.5)$$

where $u, v \in \Omega$, N is the neighborhood of $\mathbf{p}$, m and n are indices for pixels in N, and I_{k-1} and I_k are the intensity functions in images $(k-1)$ and (k).

The size of the neighborhood N must be carefully selected to ensure proper system performance. Too small an N fails to capture large displacements while too large an N enhances the background, causing inaccurate displacements, and increases the associated computational overhead. In either case, an algorithm based upon the SSD technique may fail due to inaccurate displacements.

An algorithm based upon the SSD technique may also fail due to repeated patterns in the intensity function of the image or due to large areas of uniform intensity in the image. Both cases can provide multiple matches within a feature point's neighborhood, resulting in spurious displacement measures. In order to avoid this problem, the system automatically evaluates and selects feature points.

2.2 Feature Point Selection

Feature points are selected using the SSD measure combined with an auto-correlation technique to produce an SSD surface corresponding to an auto-correlation in the neighborhood N [1] [12]. Several possible confidence measures can be applied to the surface to measure the suitability of a potential feature point.

The selection of a confidence measure is critical since many such measures lack the robustness required by changes in illumination, intensity, etc. We utilize a two-dimensional displacement parabolic fit that attempts to fit a parabola ($e(\Delta x_r) = a\Delta x_r^2 + b\Delta x_r + c$) to a cross-section of the surface derived from the SSD measure [12]. The parabola is fit to the surface in several predefined directions. Papanikolopoulos [12] selected a feature point if the minimum directional measure was sufficiently high, as measured by equation (2.5).

We extend this approach in response to the aperture problem [7]. Briefly stated, the aperture problem refers to the motion of points that lie on a feature such as an edge. Any sufficiently small local measure of motion can only retrieve that component of motion orthogonal to the edge. Any component of motion due to movement other than perpendicular to the edge is unrecoverable due to multiple matches for any given feature point.

The improved confidence measure for feature points permits the selection of a feature if it has a sufficiently high correlation with the parabola in a particular direction. For instance, an feature point corresponding to an edge will only have a high correlation in the direction perpendicular to the edge.

This directional information is then used during the depth recovery process to constrain the motion of the manipulator. The motion must be constrained such that the only displacements of the feature point occur orthogonal to the directional feature. This ensures that the only component of motion is precisely the component that can be recovered under consideration of the aperture problem. This ideal motion is not always achievable; therefore, motions will occur that are approximately, but not completely, orthogonal to the directional feature.

In response to this uncertainty in the actual motion of the manipulator, matches that lie closer to the direction of the ideal manipulator motion will be preferred if multiple matches are detected.

3 Depth Recovery Modeling

Several robotic servoing problems require accurate information regarding the depth of the object of interest. When this depth is not provided *a priori*, it must be recovered from observations of the environment via sensing. Traditional depth recovery techniques in a robotic system may suffer from various problems, including computational overhead, expensive calibration, frequent recalibration, obstruction, and solution of the correspondence problem. We present an alternative algorithm for depth recovery using the Controlled Active Vision framework.

Consider a target object at an unknown depth with a feature point, **P**, located on the surface of the object. The point **P** projects onto a camera's image plane at point **p**. By moving the visual sensor, a sequence of images and the respective projections of **P** can be effected. By observing the changes of **P**'s projections, **p**(k) and **p**$(k-1)$, in two successive images $((k)$ and $(k-1)$, respectively), an estimate of the depth of **P** is derived. Over multiple images, this estimate can be refined using adaptive techniques.

Previous work in depth recovery using active vision relied upon random camera displacements to provide displacement changes in the **p**(k)'s [4] [9] [13] [14] [15]. In our proposal, we select features automatically, produce feature trajectories (a different trajectory for every feature), and predict future displacements using the depth estimate. The errors in these predicted displacements, in conjunction with the estimated depth, are included as inputs in the control loop of the system. Thus, the next calculated movement of the system produces a camera movement that will eliminate the largest possible portion of the observed error while adhering to various environmental and manipulator specific constraints. This purposeful movement of the visual sensor provides more accurate depth estimates and a faster convergence of the depth measures over a sequence of estimates.

3.1 Formulation of Depth Recovery

The first part of our presentation focuses on the modeling of the depth recovery problem. We will use the SSD optical flow to measure displacements of feature points in the image plane. For a single feature point, **p**, we assume that the optical flow at time instant kT is $(u(kT), v(kT))$, where T is the time between two successive frames. It can be shown that at time instant kT, the optical flow is given by:

$$u(kT) = u_o(kT) + u_c(kT) \quad \text{and} \quad v(kT) = v_o(kT) + v_c(kT). \tag{3.1}$$

Since we are recovering the depth of feature points on a static object, the components of optical flow due to the movement of the object ($u_o(kT)$ *and* $v_o(kT)$) are zero. Also, equations (3.1) do not account for the computational delays involved with the calculation of the optical flow. By taking these delays into account, and by eliminating the flow components due to object motion, these equations become

$$u(kT) = u_c((k-d+1)T) \quad \text{and} \quad v(kT) = v_c((k-d+1)T) \tag{3.2}$$

where d is the computational delay factor ($d \in \{1,2,...\}$). In order to simplify the notation, the index k will be used instead of kT.

By utilizing the relations

$$u(k) = \frac{x(k+1) - x(k)}{T} \quad \text{and} \quad v(k) = \frac{y(k+1) - y(k)}{T}$$

the equations can be rewritten as:

$$x(k+1) = x(k) + Tu_c(k-d+1) \tag{3.3}$$

$$y(k+1) = y(k) + Tv_c(k-d+1)\,. \tag{3.4}$$

Furthermore, we can include the inaccuracies of the model (e.g., neglected accelerations, inaccurate robot control) as white noise. The equations (3.3) and (3.4) become

$$x(k+1) = x(k) + Tu_c(k-d+1) + v_1(k) \tag{3.5}$$

$$y(k+1) = y(k) + Tv_c(k-d+1) + v_2(k) \tag{3.6}$$

where $v_1(k)$ and $v_2(k)$ are zero mean, mutually uncorrelated, stationary random variables with variances σ_1^2 and σ_2^2 respectively.

The above equations may be written in state-space form as

$$\mathbf{p}(k+1) = \mathbf{A}(k)\,\mathbf{p}(k) + \mathbf{B}(k-d+1)\,\mathbf{u}_c(k-d+1) + \mathbf{H}(k)\,\mathbf{v}(k) \tag{3.7}$$

where $\mathbf{A}(k) = \mathbf{H}(k) = \mathbf{I}_2$, $\mathbf{p}(k)$ and $\mathbf{v}(k) \in R^2$, and $\mathbf{u}_c(k-d+1) \in R^6$. The matrix $\mathbf{B}(k) \in R^6$ is given by:

$$T\begin{bmatrix} \dfrac{-f}{Z_W(k)\,s_x} & 0 & \dfrac{x(k)}{Z_W(k)} & \dfrac{x(k)\,y(k)\,s_y}{f} & -\left(\dfrac{f}{s_x} + \dfrac{x^2(k)\,s_x}{f}\right) & \dfrac{y(k)\,s_y}{s_x} \\ 0 & \dfrac{-f}{Z_W(k)\,s_y} & \dfrac{y(k)}{Z_W(k)} & \dfrac{f}{s_y} + \dfrac{y^2(k)\,s_y}{f} & \dfrac{-x(k)\,y(k)\,s_x}{f} & -\dfrac{x(k)\,s_x}{s_y} \end{bmatrix}. \tag{3.8}$$

The vector $\mathbf{p}(k) = (x(k), y(k))^{\mathrm{T}}$ is the state vector, $\mathbf{u}_c(k-d+1) = (T_x(k-d+1), T_y(k-d+1), 0, 0, 0, 0)^{\mathrm{T}}$ is the control input vector (only movement in the x-y plane of the end-effector frame is used), and $\mathbf{v}(k) = (v_1(k), v_2(k))^{\mathrm{T}}$ is the white noise vector.

The term, $\mathbf{B}(k-d+1)\,\mathbf{u}_c(k-d+1) \in R^6$ can be simplified to

$$\mathbf{B}(k-d+1)\,\mathbf{u}_c(k-d+1) = T\begin{bmatrix} \dfrac{-f}{Z_W s_x} T_x(k-d+1) \\ \dfrac{-f}{Z_W s_y} T_y(k-d+1) \end{bmatrix} \tag{3.9}$$

since Z_W does not vary over time and only the $T_x(k-d+1)$ and $T_y(k-d+1)$ control inputs are used. By substitution into equation (3.7) and simplification, we derive the following:

$$\begin{bmatrix} x(k+1) \\ y(k+1) \end{bmatrix} = \begin{bmatrix} x(k) \\ y(k) \end{bmatrix} + T\begin{bmatrix} \dfrac{-f}{Z_W s_x} T_x(k-d+1) \\ \dfrac{-f}{Z_W s_y} T_y(k-d+1) \end{bmatrix} + \begin{bmatrix} v_1(k) \\ v_2(k) \end{bmatrix}. \tag{3.10}$$

The previous equation is used under the assumption that the optical flow induced by the motion of the camera does not change significantly in the time interval T. Additionally, the interval T should be as small as possible so that the relations that provide $T_x(k-d-1)$ and $T_y(k-d+1)$ are as accurate as possible. The parameter T has as its lower bound 16 msec (the sampling rate of standard video equipment). If the upper bound on T can be reduced such that it is lower than the sampling rate of the manipulator controller (in our case, 28 msec for the PUMA 560), then the system will no longer be constrained by the speed of the vision hardware.

The objective is to design a specific trajectory (a set of desired $\mathbf{p}^*(k)$

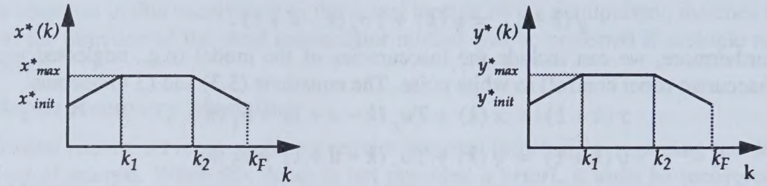


Figure 2: Desired trajectories of the features

$= (x^*(k), y^*(k))^{\mathrm{T}}$ for successive k's) for every individual feature in order to identify the depth parameter Z_W. Thus, we have to design a control law that forces the eye-in-hand system to track the desired trajectory for every feature. Based on the information that comes out of the automatic feature selection procedure we select a trajectory that for a corner feature is depicted in Figure 2. Then, the control objective function is selected to be

$$\begin{aligned} F_o(k+d) = E\{ & (\mathbf{p}_m(k+d)-\mathbf{p}^*(k+d))^{\mathrm{T}}\,\mathbf{Q} \\ & (\mathbf{p}_m(k+d)-\mathbf{p}^*(k+d)) + \mathbf{u}_{\mathrm{c}}^{\mathrm{T}}(k)\,\mathbf{L}_{\mathrm{d}}\mathbf{u}_{\mathrm{c}}(k)\} \end{aligned} \tag{3.11}$$

where $E\{Y\}$ denotes the expected value of the random variable Y, $\mathbf{p}_m(k)$ is the measured state vector, and $\mathbf{Q}$, $\mathbf{L}_{\mathrm{d}}$ are control weighting matrices. Based on the equation (3.10) and the minimization of the control objective function $F_o(k+d)$ (with respect to the vector $\mathbf{u}_{\mathrm{c}}(k)$), we can derive the following control law

$$\begin{aligned} \mathbf{u}_{\mathrm{c}}(k) = & -\left[\hat{\mathbf{B}}^{\mathrm{T}}(k)\,\mathbf{Q}\hat{\mathbf{B}}(k) + \mathbf{L}_{\mathrm{d}}\right]^{-1}\hat{\mathbf{B}}^{\mathrm{T}}(k)\,\mathbf{Q} \\ & \left\{(\mathbf{p}_m(k)-\mathbf{p}^*(k+d)) + \sum_{i=1}^{i=d-1}\hat{\mathbf{B}}(k-i)\,\mathbf{u}_{\mathrm{c}}(k-i)\right\} \end{aligned} \tag{3.12}$$

where $\hat{\mathbf{B}}(k)$ is the estimated value of the matrix $\mathbf{B}(k)$. The matrix $\hat{\mathbf{B}}(k)$ depends on the estimated value of the depth $\hat{Z}_W$. The estimation of the depth parameter is performed by using the procedure described in [12]. In order to use this procedure, we must rewrite equation (3.10) as follows (for one feature point):

$$\Delta\mathbf{p}_m(\mathrm{k}) = \zeta_W\,\mathbf{u}_{\mathrm{cnew}}(k-d) + \mathbf{n}(k) \tag{3.13}$$

where

$$\Delta\mathbf{p}_m(\mathrm{k}) = \mathbf{p}_m(\mathrm{k}) - \mathbf{p}_m(\mathrm{k}-1)$$

$$\zeta_W = f/(s_x Z_W)$$

$$\mathbf{n}(k) \sim N(0,\mathbf{N}(k))$$

$$\mathbf{u}_{\mathrm{cnew}}(k) = T\begin{bmatrix} -T_x(k) \\ -\dfrac{s_x}{s_y}T_y(k) \end{bmatrix}.$$

The objective of the estimation scheme is to estimate the parameter ζ_W for each individual feature. When this computation is completed, it is trivial to compute the parameter $1/Z_W$ that is needed for the construction of the depth maps. It is assumed that the matrix $\mathbf{N}(k)$ is constant ($\mathbf{N}(k) = \mathbf{N}$). We use the estimation scheme described in [8] [10] (for one feature point)

$$ {}^{p}\hat{\zeta}_{W}(k) = {}^{u}\hat{\zeta}_{W}(k-1) \tag{3.14} $$

$$ {}^{p}pr(k) = {}^{u}pr(k-1) + s(k-1) \tag{3.15} $$

$$ {}^{u}pr(k) = \{ ({}^{p}pr(k))^{-1} + \mathbf{u}^{\mathrm{T}}_{\mathrm{cnew}}(k-d)\,\mathbf{N}^{-1}(\mathbf{u}_{\mathrm{cnew}}(k-d)) \}^{-1} \tag{3.16} $$

$$ \mathbf{g}^{T}(k) = {}^{u}pr(k)\,\mathbf{u}^{T}_{\mathrm{cnew}}(k-d)\,\mathbf{N}^{-1} \tag{3.17} $$

$$ {}^{u}\hat{\zeta}_{W}(k) = {}^{p}\hat{\zeta}_{W}(k) + \mathbf{g}^{T}(k)\,\{\Delta\mathbf{p}_{m}(k) - {}^{p}\hat{\zeta}_{W}(k)\,\mathbf{u}_{\mathrm{cnew}}(k-d)\} \tag{3.18} $$

where the superscript *p* denotes the predicted value of a variable, the superscript *u* denotes the updated value of a variable, and *s (k)* is a covariance scalar. The initial conditions are described in [12]. The term ${}^{p}pr(0)$ can be viewed as a measure of the confidence that we have in the initial estimate ${}^{p}\hat{\zeta}_{W}(0)$. The next section describes several simulation and experimental results using this scheme.

4 Simulation and Experiments

4.1 Simulation Results

The model was initially verified in simulation experiments. Simulations were performed on a Silicon Graphics Indigo™. The camera and manipulator were modeled, as was the target and the environment. The target was displayed in depth and was projected using a pinhole camera model. It appeared in front of a regular textured background that was placed at optical infinity.

The camera was modeled using the perspective transformation of the Indigo's geometry engine. The camera model was based upon a 7.5 mm focal length lens and an image plane that was sized to correspond to a 1/2″ CCD camera array. The image plane was placed in front of the nodal point of the optical system due to the modeling specifically used by the Indigo.

The manipulator was modeled using second-order partial differential equations that transform the ideal control vector into a realized control vector based upon the current motion vector of the manipulator.

The initial simulation runs were conducted with a soda-can-sized cylinder placed 45 cm in depth. The controller began the depth extraction process with an initial depth parameter estimate of 1 meter. The recovery process consisted of a sequence of motions, each producing an image. Once completed, the recovered depth of each feature point was compared to the actual depth to produce a measure of the accuracy of the process. For virtually all feature points, the variation of the recovered versus the computed was subpixel in nature. In fact, the majority of the sub-pixel errors were less than one half of the error calculated for a given displacement error of one pixel. It should be noted that sub-pixel fitting and multi-grid methods were not employed in this process due to their computational overhead. The reconstructed surface is shown on the left in Figure 3.

A second set of simulation runs were conducted using two soda-can-sized cylinders. The first cylinder was rendered at 45 cm in depth. The second was rendered at 70 cm in depth. Again, a regular textured background was placed at optical infinity. The initial depth parameter estimate was set to 1 meter. The results of the process were again highly accurate, achieving sub-pixel accuracy in most cases. The result of the surface reconstruction is shown on the right in Figure 3. The reconstruction retains the perspective effect upon the apparent relative size of the targets; thus, the target at greater depth appears smaller.

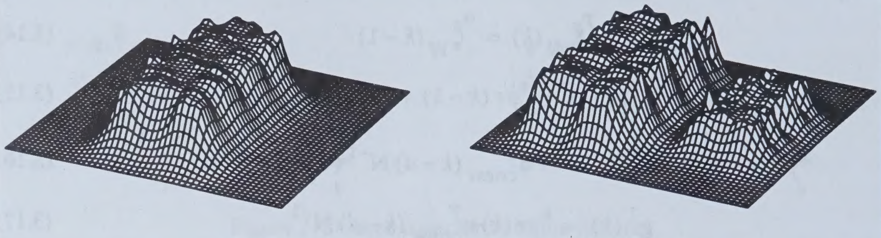


Figure 3: Single and dual target surfaces from simulation

4.2 Hardware Results

Once we determined that the performance of the simulation was satisfactory, the algorithm was implemented on the Minnesota Robotic Visual Tracker (MRVT) system (see Figure 4). The system consists of a PUMA 560 manipulator controlled via the Unimate controller's Alter line. The Alter line requires path control updates once every 28 msec. Those updates are provided by an Ironics 68030 VME processor that shares its bus with a Sun Sparc 330 and a Datacube vision system via BIT-3 bus extenders. A Panasonic miniature camera is mounted parallel to the end-effector of the PUMA and provides its output to the Datacube for processing. The Datacube system consists of a MVME-147 processor, a MaxVideo 20 video processor, and a Max860 vector processor. The vision sub-system performs the optical flow, calculates the input control vector according to the control law, and supplies the input vector to the Ironics processor for transmission to the Unimate controller.

The initial experimental runs were conducted with a soda can placed in depth such that the leading edge of the curved surface was 53 cm from the nodal point of the camera (see Figure 5). Similar to the simulation experiments, the initial depth estimate was set to 1 meter of depth. The modelling for the visual process was precisely the same as the simulation runs; however, the camera scaling factors and the controller gains were changed to meet the specifications of the miniature camera and the PUMA manipulator.

Again, the resulting data for the feature points was not subjected to sub-pixel fitting nor

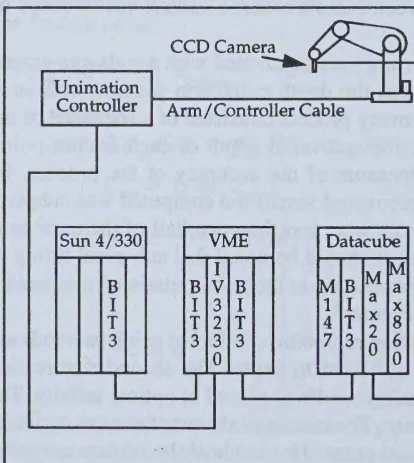


Figure 4: MRVT system architecture

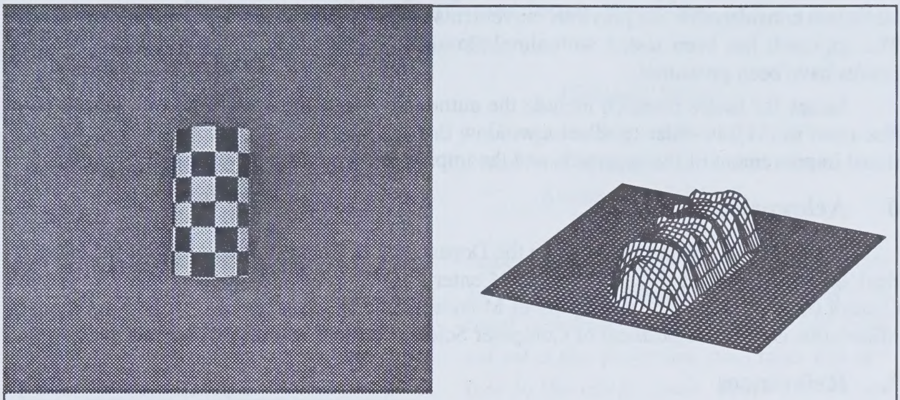


Figure 5: Single can target and surface

multi-grid methods. It should be noted that, similar to the simulation, the error in the recovered depth of the majority of feature points was sub-pixel in nature. The reconstructed surface of the single soda can is shown in Figure 5.

A second set of experiments was also conducted in the same fashion as the simulations, using two soda cans as targets. The leading edges of the cans were placed at 53 cm and 41 cm in depth (see Figure 6). Again, the majority of the errors in the calculated depths were sub-pixel. For these runs, the initial depth estimate was an underestimate of 25 cm, in contrast to the previous simulations and experiments that used overestimates. The reconstructed surfaces of the dual can experiment are shown in Figure 6.

5 Conclusion

This paper presents a novel technique for computing depth maps through controlled active exploration with a camera. Unlike similar approaches [6] [8] [9] [13] [14], we propose a scheme that is based on automatic selection of features and design of specific trajectories on the image plane for each individual feature. This approach helps us design trajectories that provide maximum identifiability of the depth parameter. During the execution of the specific trajectory, the depth parameter is computed with the help of a simple estimation scheme that

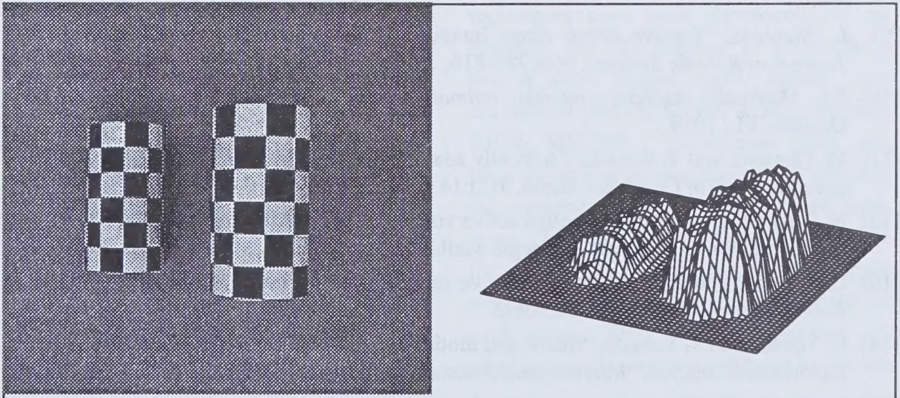


Figure 6: Two can target and surface

takes into consideration the previous movements of the camera and the computational delays. The approach has been tested with simulations and experiments, and several experimental results have been presented.

Issues for future research include the automatic selection of the window size (an issue discussed in [11]) in order to select a window that has some texture variations, the computational improvement of the approach, and the implicit incorporation of the robot dynamics.

6 Acknowledgments

This work has been supported by the Department of Energy (Sandia National Laboratories) through Contract #AC-3752D, the Center for Advanced Manufacturing, Design and Control (CAMDAC) of the University of Minnesota, the Graduate School of the University of Minnesota, and the Department of Computer Science at the University of Minnesota.

7 References

[1] P. Anandan, "Measuring visual motion from image sequences," Technical Report COINS-TR-87-21, COINS Department, University of Massachusetts, 1987.

[2] J.J. Craig, *Introduction to robotics: mechanics and control*, Addison-Wesley, Reading, MA, 1985.

[3] J.T. Feddema and C.S.G. Lee, "Adaptive image feature prediction and control for visual tracking with a hand-eye coordinated camera," *IEEE Transactions on Systems, Man, and Cybernetics*, 20(5):1172-1183, 1990.

[4] C.P. Jerian and R. Jain, "Structure from motion—a critical analysis of methods," *IEEE Transactions on Systems, Man, and Cybernetics*, 21(3):572-588, 1991.

[5] J. Heel, "Dynamic motion vision," *Robotic and Autonomous Systems*, 6(3):297-314, 1990.

[6] K.N. Kutulakos and C.R. Dyer, "Recovering shape by purposive viewpoint adjustment," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 16-22, 1992.

[7] D. Marr, *Vision*, W.H. Freeman and Company, San Francisco, CA, 1982.

[8] L. Matthies, T. Kanade, and R. Szeliski, "Kalman filter-based algorithms for estimating depth from image sequences," *International Journal of Computer Vision*, 3(3):209-238, 1989.

[9] L. Matthies, "Passive stereo range imaging for semi-autonomous land navigation," *Journal of Robotic Systems*, 9(6):787-816, 1992.

[10] P.S. Maybeck, *Stochastic models, estimation, and control*, Vol. 1, Academic Press, Orlando, FL, 1979.

[11] M. Okutomi and T. Kanade, "A locally adaptive window for signal matching," *International Journal of Computer Vision*, 7(2):143-162, 1992.

[12] N. Papanikolopoulos, "Controlled active vision," Ph.D. Thesis, Department of Electrical and Computer Engineering, Carnegie Mellon University, 1992.

[13] C. Shekhar and R. Chellappa, "Passive ranging using a moving camera," *Journal of Robotic Systems*, 9(6):729-752, 1992.

[14] C. Tomasi and T. Kanade, "Shape and motion from image streams under orthography: a factorization method," *International Journal of Computer Vision*, 9(2):137-154, 1992.

[15] D. Weinshall, "Direct computation of qualitative 3-D shape and motion invariants," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 13(12):1236-1240, 1991.

Reconstructing Surfaces from Profiles: Epipolar Curves

Richard Szeliski
Digital Equipment Corporation,
Cambridge Research Lab,
One Kendall Square, Bldg. 700
Cambridge, MA 02139

Richard Weiss
Computer Science Department
University of Massachusetts at Amherst
Amherst, MA 01003

Abstract

Recovering the shape of an object from two views, or binocular stereo, fails at occluding contours of smooth objects because the extremal contours are view dependent. For three or more views, shape recovery is possible, and several algorithms have recently been developed for this purpose. We present a new approach to the multiframe stereo problem which does not depend on differential measurements in the image, which may be noise sensitive. Instead, we use a linear smoother to optimally combine all of the measurements available at the contours (and other edges) in all of the images. This allows us to extract a robust and dense estimate of surface shape, and to integrate shape information from both surface markings and occluding contours.

1 Introduction

Most visually-guided systems require representations of surfaces in the environment in order to integrate sensing, planning, and action. The task considered in this paper is the recovery of 3D structure (shape) of objects with piecewise-smooth surfaces from a sequence of profiles taken with known camera motion. The *profile* (also known as the *extremal boundary* or *occluding contour*) is defined as the image of the *critical set* of the projection map from the surface to the image plane. Since profiles are general curves in the plane without distinguished points, there is no *a priori* pointwise correspondence between these curves in different views. However, given the camera motion, there is a correspondence based on the *epipolar constraint.* For two images, i.e., classical *binocular stereo*, this epipolar constraint is a set of straight lines which are the intersection of the *epipolar planes* with the image plane. The epipolar plane through a point is determined by the view direction at that point and the instantaneous camera translation direction.

In the case of contours that are not view dependent, e.g., creases (tangent discontinuities) and surface markings, many techniques have been developed for recovering the 3D contour locations from two or more images under known camera motion [MP79, MF80, Arn83, BBM87, BB89, MKS89]. Techniques have also been developed for simultaneously estimating contour locations and camera positions [TH84, FLT87, Hor90]. However, for smooth curved surfaces, the critical set which generates the profile is different for each view. Thus, the triangulation applied in two-frame stereo will not be correct

---

[*] The second author acknowledges the support of grants TACOM DAAE07-91-C-RO35, NSF IRI-9208920, and NSF IRI-9116297.

along the occluding contour for smooth surfaces. On the other hand, the fact that the critical sets sweep out an area means that the connectivity of the surface points can be determined, i.e., one obtains a surface patch rather than a set of points.

The problem of reconstructing a smooth surface from its profiles has been explored for known planar motion by Giblin and Weiss [GW87] and subsequently for more general known motion by Vaillant and Faugeras [Vai90, VF92] and Cipolla and Blake [BC90, CB90, CB92]. These approaches are either based on a differential formulation and analysis, or they use curve fitting but still only use three frames. First order temporal derivatives are usually computed as differences from pairs of frames, and second order derivatives from triples. Unfortunately, determining differential quantities reliably from real images in this way is difficult. Even fitting curves to data from three images can be unsatisfactory. This has led Cipolla and Blake to use relative measurements in order to cancel some of the error due to inadvertent camera rotation. Their approach approximated image contours with B-snakes which require initialization for each contour that is tracked. In addition, B-snakes implicitly smooth the contours in the image. Since the recovery of 3D points is a linear problem (as we will show in this paper), the smoothing can be done in 3D on the surface, where more context can be used in the detection of discontinuities so that detailed structure can be preserved.

It is natural to consider surface reconstruction as an optimal estimation problem. To overcome the limitations of previous algorithms, the approach we develop in this paper applies standard techniques from estimation theory (Kalman filtering and smoothing) to make optimal use of each measurement without computing differential quantities. First, we derive a *linear* set of equations between the unknown shape (surface point positions and radii of curvature) and the measurements. We then develop a robust linear smoother ([Gel74, Bie77]) to compute statistically optimal current and past estimates from the set of contours. Smoothing allows us to combine measurements on both sides of each surface point.

Our technique produces a complete surface description, i.e., a network of linked 3D surface points, which provides us with a much richer description than just a set of 3D curves. Due to self-occlusion and occlusion by other surfaces, some parts of the surface may never appear on the profile. Since the method presented here also works for arbitrary surface markings and creases, a larger part of the surface can be reconstructed than from occluding contours of the smooth pieces alone. Our approach also addresses the difficult problem of contours that merge and split in the image, which must be resolved if an accurate and complete 3D surface model is to be constructed.

The method we develop has applications in many areas of computer vision, computer aided design, and visual communications. The most traditional application of visually based shape recovery is in the reconstruction of a mobile robot's environment, which allows it to perform obstacle avoidance and planning tasks [CBZ92]. Visually based shape recovery can also be used to develop strategies for robotics grasping and manipulation tasks, or as an off-line technique to automatically "learn" object descriptions for object or pose recognition tasks. In less traditional applications, our system could be used to perform *reverse engineering*, i.e., to automatically acquire 3D computer aided design (CAD) description of real-world objects or prototypes, or even to construct a "3D fax" for transmitting 3D object descriptions and images between desktops.

2 Reconstructing surface patches

The surface being reconstructed from a moving camera can be parametrized in a natural way by two families of curves [GW87, CB90]: one family consists of the critical sets on the surface; the other is tangent to the family of rays from the camera focal points. These two families of rays are transverse except on a set of measure zero, e.g. when the profile is singular. The latter curves are called *epipolar curves.* For a pair of stereo images, each viewing direction together with the vector from one camera center to the other determines a plane called an *epipolar plane.* The same construction holds in the case of motion: in the limit, as the time between samples goes to zero, the plane determined by a view direction and the camera translation velocity will also be called an epipolar plane. Another way to look at the epipolar curves is that for each camera position, there is a tangent vector field at every point on the critical set. For each camera trajectory, this tangent field is defined over an area of the surface. Since a point may be on more than one critical set at different times, the tangent field may have multiple values. Nevertheless, locally one can define the integral curves to each branch of this tangent field as the epipolar curves. Thus, the epipolar curves are always tangent to the view direction.

Let (u, t) be a local parametrization of the surface such that t = constant are the critical sets and u = constant are the epipolar curves. Let $r(u,t)$ be a point on the surface, then the reconstruction formula is

$$r(u,t) = c(t) + \lambda(u,t)p(u,t) \qquad (1)$$

where c is the camera position, λ is the distance from the camera center to r, and p is a unit vector denoting the viewing direction to the point on the surface. The condition the epipolar curves are given by u = constant is given in terms of the derivative with respect to t by $r_t \| p$. Even if the parametrization does not satisfy this last constraint, there is a reconstruction formula given by

$$\lambda(u,t) = \frac{-c_t(t) \cdot n(u,t)}{p_t(u,t) \cdot n(u,t)} \qquad (2)$$

where n is a normal vector to the surface, which can be computed from the profile p. The problem is that this formula only gives a first order approximation to the surface. In order to get a higher order approximation, three or more images are needed.

Cipolla and Blake [CB90, CB92] and Vaillant and Faugeras [Vai90, VF92] noticed that to compute the curvature of a planar curve from three tangent rays, one can determine a circle which is tangent to these rays. The assumption that one needs to make is that the surface remains on the same side of the tangent rays. This is true on intervals of the curve for which the tangent does not have order of contact with the surface greater than two. This corresponds to a singularity of the profile. In addition, the epipolar curve can have a singularity when the epipolar plane is tangent to the surface or can end where the critical set is occluded as at a T-junction [GW93].

Given three or more edgels tracked with our technique, we would like to compute the location of the surface and its curvature by fitting a circular arc to the lines defined by the view directions at those edgels. In general, a non-singular space curve will have a unique circle which is closest to the curve at any given point. This is called the *osculating circle*, and the plane of this circle is called the *osculating plane*. It is easy to see that if the epipolar curve is non-singular, then the epipolar plane is an estimate of its osculating plane [CB92], and the lines defined by the view directions can be projected onto this plane.

The relationship between the curvature of a curve such as the epipolar curve and the curvature of the surface is determined by the angle between the normal to the curve and the normal to the surface. The curvature of the curve scaled by the cosine of this angle is the *normal curvature*. The curvature of a surface can be thought of as a function which assigns to every tangent direction v a value which is the curvature k_v of the normal slice in that direction. If v is the tangent to the epipolar curve, k_{epi} is the curvature of the epipolar curve, and ϕ is the angle between the epipolar plane and the plane containing v and the surface normal, then the relationship among them is given by the equation

$$k_v = k_{epi} \cos \phi \tag{3}$$

This gives Meusnier's Theorem which says that the normal curvature is the same for all curves on the surface with a given tangent direction. Since the normal to the surface can be determined from the image, the normal curvature can be obtained from the epipolar curve.

3 Measurement equations

Once we have selected the reconstruction plane for fitting the circular arc to be the epipolar plane, we must compute the set of lines in this plane which should be tangent to the circle. This can be done either by projecting the 3D lines corresponding to the linked edgels directly onto the plane, or by intersecting the tangent planes (defined by the edgels and their orientations) with the reconstruction plane.

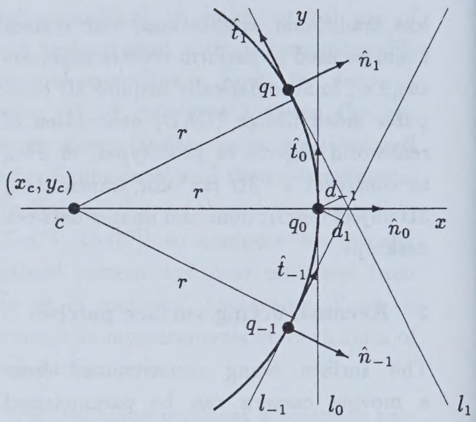


Figure 1: Local coordinate axes and circle center point calculation

In this figure, the points p_i and q_i are coincident. In general, the q_i will lie somewhere along the tangent rays, and the p_i will be the points of tangency to the osculating circle.

We represent the 3D line corresponding to an edgel in frame i by a 3D point q_i and a direction $\hat{t}_i$. The point q_i is chosen to be the intersection of the viewing ray with a reference plane $z = z_0$. The direction is given by $\hat{t}_i = \mathcal{N}(q_i - c_i)$, where c_i is the camera center and $\mathcal{N}()$ normalizes a vector. We choose one of these lines as the *reference frame* $(\hat{n}_0, \hat{t}_0)$ centered at q_0 (where $\hat{n}_0 = \hat{t}_0 \times \hat{n}_{epi}$), e.g., by selecting the middle of n frames for a batch fit, or the last frame for a Kalman filter. This line lies in the reconstruction plane defined by $\hat{n}_{epi}$.

If we parameterize the osculating circle by its center $c = (x_c, y_c)$ and radius r (Figure 1), we find that the tangency condition between line i and the circle can be written as

$$c_i x_c + s_i y_c + r = d_i \qquad (4)$$

where $c_i = \hat{t}_i \cdot \hat{t}_0$, $s_i = -\hat{t}_i \cdot \hat{n}_0$, and $d_i = (q_i - q_0) \cdot \hat{n}_i$. *Thus, we have a linear estimation problem in the quantities* (x_c, y_c, r) *given the known measurements* (c_i, s_i, d_i). This linearity is central to the further developments in the paper, including the least squares fitting, Kalman filter, and linear smoother, which we develop in the next three sections.

4 Least squares fitting

While in theory the equation of the osculating circle can be recovered given the projection of three non-parallel tangent lines onto the epipolar plane, a much more reliable estimate can be obtained by using more views. Given the set of equations (4), how can we recover the best estimate for (x_c, y_c, r)? Regression theory [Alb72, Bie77] tells us that the minimum least squared error estimate of the system of equations $Ax = d$ can be found by minimizing

$$e = |Ax - d|^2 = \sum_i (a_i \cdot x - d_i)^2. \qquad (5)$$

5 Experimental results

To determine the performance of our shape reconstruction algorithm, we generated a synthetic motion sequence of a truncated ellipsoid rotating about its z axis. The camera was oblique (rather than perpendicular) to the rotation axis, so that the epipolar curves are not planar, and the reconstruction plane is continuously varying over time. We chose to use a truncated ellipsoid since it is easy to analytically compute its projections (which are ellipses, even under perspective), and since its radius of curvature is continuously varying (unlike, a cylinder).

When we run the edge images through our least-squares fitter or Kalman filter/smoother, we obtain a series of 3D curves. The curves corresponding to the surface markings and ridges (where the ellipsoid is truncated) should be stationary and have 0 radius, while the curves corresponding to the occluding contour should continuously sweep over the surface.

The results of the reconstruction can be found in [SW93]. To obtain a quantitative measure of the reconstruction algorithm performance, we computed the root median square error between the reconstructed 3D coordinates and the true 3D coordinates (which are known to the synthetic sequence generating program). The reconstruction error for a window size of seven frames was on the order of 0.1% when noise with a standard deviation of 0.1 pixels was added. By being more selective about which 3D estimates are considered valid (either by requiring more frames to have been successfully fit, or lowering the threshold on maximum covariance), a more reliable estimate can be obtained at the expense of fewer recovered points.

We have also applied our algorithm to the four real image sequences shown in Figure 2. These sequences were obtained by placing an object on a rotating mechanized turntable whose edge has a Gray code strip used for reading back the rotation angle [Sze91, Sze93]. Figure 3 shows the two sets of reconstructed 3D curves. We can see that the overall shape of the objects has been reconstructed quite well. We show only the profile curves, since the epipolar curves would make the line drawing too dense for viewing at this resolution.

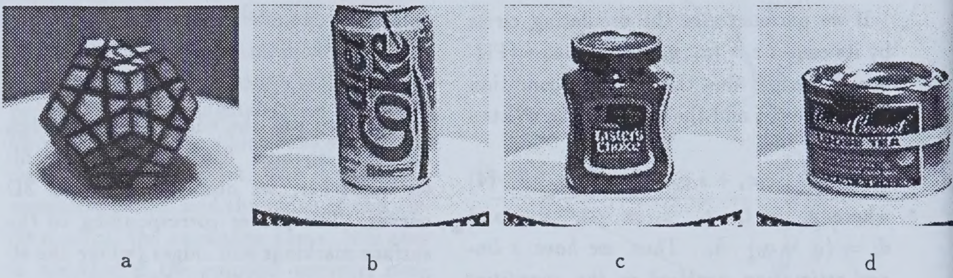


Figure 2: Sample real image sequences used for experiments: (a) dodecahedron (b) diet coke (c) coffee (d) tea.

6 Discussion and Conclusion

This paper extends previous work on both the reconstruction of smooth surfaces from profiles (edge-based multiframe stereo) and on the epipolar analysis on spatiotemporal surfaces. The ultimate goal of our work is the construction a complete detailed geometric and topological model of a surface from a sequence of views together with an estimate of uncertainty. Towards this end, our observations are connected by tracking edges over time as well as linking neighboring edges into contours. The information represented at each point includes the position, surface normal, and curvatures (currently only in the viewing direction). In addition, error estimates are also computed for these quantities. Since the sensed data does not provide a complete picture of the surface, e.g., there can be self-occlusion or parts may be missed due to coarse sampling, it is necessary to build partial models. In the context of active sensing and real-time reactive systems, the reconstruction needs to be incremental as well.

Because our equations for the reconstruction algorithm are linear with respect to the measurements, it is possible to apply statistical linear smoothing techniques, as we have demonstrated. This satisfies the requirement for incremental modeling, and provides the error estimates which are needed for integration with other sensory data, both visual and tactile. The application of statistical methods has the advantage of providing a sound theoretical basis for sensor integration and for the reconstruction process in general [Sze89, CY90].

In future work, we intend to develop a more complete and detailed surface model by combining our technique with regularization-based curve and surface models. We also plan to investigate the integration of our edge-based multiframe reconstruction technique with other visual and tactile techniques for shape recovery.

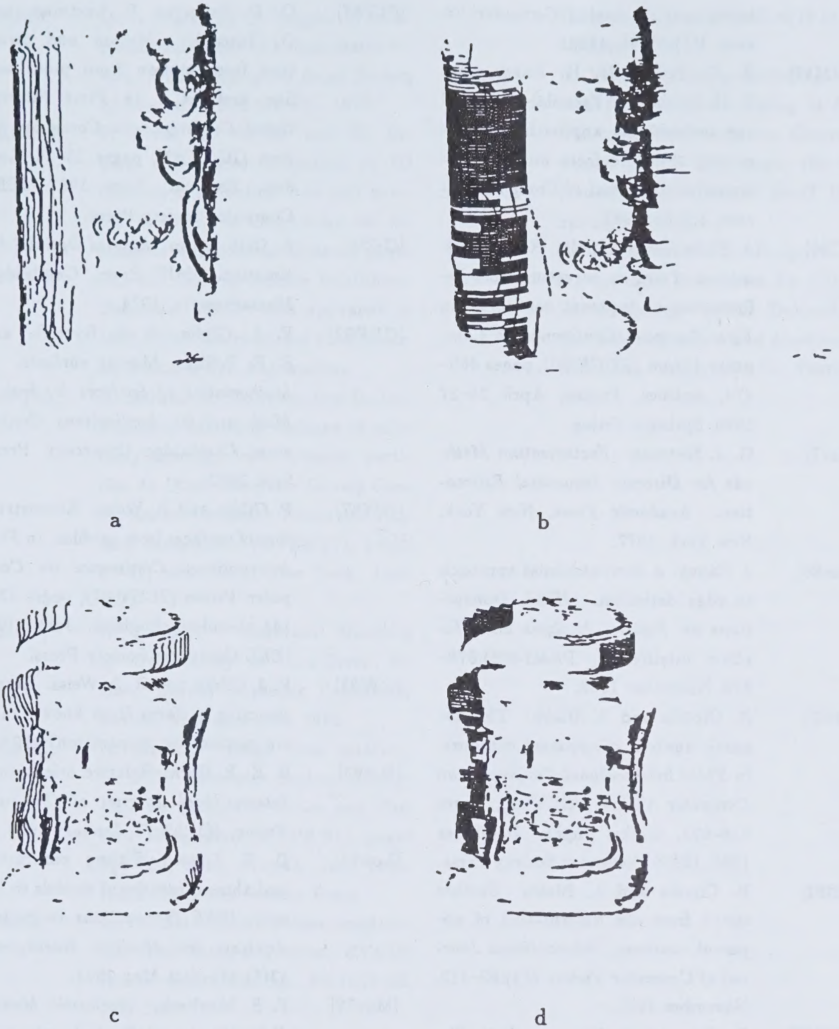
References

[Alb72] A. Albert. *Regression and the Moore-Penrose Pseudoinverse.* Academic Press, New York, 1972.

[Arn83] R. D. Arnold. Automated stereo perception. Technical Report AIM-351, Artificial Intelligence Laboratory, Stanford University, March 1983.

[Bak89] H. H. Baker. Building surfaces of evolution: The weaving wall. *International Journal of Computer Vision*, 3(1):50–71, 1989.

[BB89] H. H. Baker and R. C. Bolles. Generalizing epipolar-plane image analysis on the spatiotemporal surface. *In-*



a

b

c

d

Figure 9: Profile and epipolar curves for: (a–b) diet coke (c–d) coffee.

ternational Journal of Computer Vision, 3(1):33–49, 1989.

[BBM87] R. C. Bolles, H. H. Baker, and D. H. Marimont. Epipolar-plane image analysis: An approach to determining structure from motion. *International Journal of Computer Vision*, 1:7–55, 1987.

[BC90] A. Blake and R. Cipolla. Robust estimation of surface curvature from deformation of apparent contours. In *First European Conference on Computer Vision (ECCV'90)*, pages 465–474, Antibes, France, April 23–27 1990. Springer-Verlag.

[Bie77] G. J. Bierman. *Factorization Methods for Discrete Sequential Estimation.* Academic Press, New York, New York, 1977.

[Can86] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679–698, November 1986.

[CB90] R. Cipolla and A. Blake. The dynamic analysis of apparent contours. In *Third International Conference on Computer Vision (ICCV'90)*, pages 616–623, Osaka, Japan, December 1990. IEEE Computer Society Press.

[CB92] R. Cipolla and A. Blake. Surface shape from the deformation of apparent contours. *International Journal of Computer Vision*, 9(2):83–112, November 1992.

[CBZ92] R. Curwen, A. Blake, and A. Zisserman. Real-time visual tracking for surveillance and path planning. In *Second European Conference on Computer Vision (ECCV'92)*, pages 879–883, Santa Margherita Liguere, Italy, May 1992. Springer-Verlag.

[CY90] J. J. Clark and A. L. Yuille. *Data Fusion for Sensory Information Processing Systems.* Kluwer Academic Publishers, Boston, Massachusetts, 1990.

[FLT87] O. D. Faugeras, F. Lustman, and G. Toscani. Motion and structure from motion from point and line matches. In *First International Conference on Computer Vision (ICCV'87)*, pages 25–34, London, England, June 1987. IEEE Computer Society Press.

[Gel74] A. Gelb, editor. *Applied Optimal Estimation.* MIT Press, Cambridge, Massachusetts, 1974.

[GRP92] P. J. Giblin, J. E. Rycroft, and F. E. Pollick. Moving surfaces. In *Mathematics of Surfaces V, Inst of Math and its Applications Conference.* Cambridge University Press, Sept 1992.

[GW87] P. Giblin and R. Weiss. Reconstruction of surfaces from profiles. In *First International Conference on Computer Vision (ICCV'87)*, pages 136–144, London, England, June 1987. IEEE Computer Society Press.

[GW93] P. J. Giblin and R. S. Weiss. Reconstructing surfaces from known camera motion. (in preparation) 1993.

[Hor90] B. K. P. Horn. Relative orientation. *International Journal of Computer Vision*, 4(1):59–78, January 1990.

[Low91] D. G. Lowe. Fitting parameterized three-dimensional models to images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(5):441–450, May 1991.

[May79] P. S. Maybeck. *Stochastic Models, Estimation, and Control*, volume 1. Academic Press, New York, New York, 1979.

[MF80] J. E. W. Mayhew and J. P. Frisby. The computation of binocular edges. *Perception*, 9:69–87, 1980.

[MKS89] L. H. Matthies, T. Kanade, and R. Szeliski. Kalman filter-based algorithms for estimating depth from image sequences. *International Journal of Computer Vision*, 3:209–236, 1989.

[MP79] D. C. Marr and T. Poggio. A computational theory of human stereo vision. *Proceedings of the Royal Society of London*, B 204:301–328, 1979.

[RBE86] P. Rives, E. Breuil, and B. Espiau. Recursive estimation of 3D features using optical flow and camera motion. In *Conference on Intelligent Autonomous Systems*, pages 522–532. Elsevier Science Publishers, December 1986. Also appeared in 1987 IEEE International Conference on Robotics and Automation.

[STT93] R. Szeliski, D. Tonnesen, and D. Terzopoulos. Modeling surfaces of arbitrary topology with dynamic particles. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'93)*, pages 82–87, New York, New York, June 1993.

[Sze89] R. Szeliski. *Bayesian Modeling of Uncertainty in Low-Level Vision.* Kluwer Academic Publishers, Boston, Massachusetts, 1989.

[Sze91] R. Szeliski. Shape from rotation. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'91)*, pages 625–630, Maui, Hawaii, June 1991. IEEE Computer Society Press.

[Sze93] R. Szeliski. Rapid octree construction from image sequences. *CVGIP: Image Understanding*, 58(1):23–32, July 1993.

[SW93] R. Szeliski and R. Weiss. Robust Shape Recovery from Occluding Contours Using a Linear Smoother. *CVPR'93*, pages 666–667, New York, June 1993. IEEE Computer Society Press.

[TH84] R. Y. Tsai and T. S. Huang. Uniqueness and estimation of three-dimensional motion parameters of rigid objects with curved surfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6(1):13–27, January 1984.

[Vai90] R. Vaillant. Using occluding contours for 3D object modeling. In *First European Conference on Computer Vision (ECCV'90)*, pages 454–464, Antibes, France, April 23–27 1990. Springer-Verlag.

[VF92] R. Vaillant and O. D.. Faugeras. Using extremal boundaries for 3-D object modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):157–173, February 1992.

STABILIZATION OF TIME - DELAY CONTROL SYSTEMS USING CAUCHY MATRIX

A.S.C. Sinha
Purdue University at Indianapolis
723 W. Michigan Street
Indianapolis, IN 46202-5132

Abstract

The problem of stabilizing a linear time-delay control systems is solved by using Cauchy matrix. A constructive algorithm is formulated for constructing a Cauchy matrix of a linear control system with a variable delay by using the fundamental matrix of the corresponding ordinary differential equations. The method is illustrated by the construction of a Cauchy matrix of a second order system.

Introduction

Recently many authors have dealt with linear functional differential equations and obtained many results similar to those in linear ordinary differential equations. It is well-known that the set of solutions of linear functional differential equation systems is a linear space. However, unfortunately this linear space is in general, of infinite dimension. On the other hand, Hale [1] has shown that for the linear system, there is an nxn-matrix C(t,s) which is a solution of the homogeneous linear system through the initial function ϕ at $t=t_0$ such that $C(s,s)=I$, $C(s + \theta; s)=0$ for $\theta < 0$, where C(t,s) is called "Cauchy matrix" which is different from a corresponding matrix of ordinary differential equation. The structure of Cauchy matrix for delay differential system is more complex and it has essential singularities, whereas the fundamental matrix of ordinary differential equation is nonsingular, whatever the values of the arguments t and s.

Until recently, most authors have devoted their research to the theory of perturbed control systems described by linear differential equations with continuous perturbations. [2-4]. The present paper deals with a class of perturbed control systems with lags by an analytic method by constructing a Cauchy matrix for the delay-differential system. We give a new method of stabilizing and constructing a Cauchy matrix C(t,s) for the delay-differential control system.

Main Result

Let ***C*** be the space of all continious R^n - valued functions defined on $[-\tau,0]$ for a given $\tau>0$, and let $|\phi|$ be the norm in ***C*** given by $|\phi| = \sup \{|\phi(\theta)| ; 0\geq\theta\geq-\tau\}$. For a continious R^n- valued function x(s), x_t denotes the function in ***C*** such that $x_t(\theta) = x(t+\theta)$ for $\theta \in [-\tau,0]$ and x(t) denotes the right-hand derivative.

Consider the system of the form

$$\dot{x}(t) = \sum_{k=0}^{1} [A_k + B_k(t)]\, x(t-\tau_k(t)) + q_1 u_1 + \ldots + q_m u_m \tag{1}$$

where it is assumed that $0 \equiv \tau_0(t) < \tau_1(t)$. The matrix $B_k(t)$ is assumed continuous and locally summable for each k. q_i (i=1, ...,m) are constant vectorsin R^n. Let $Q=(q_1, \ldots, q_m)$. If we choose the control function $u \in R^m$ according to the algorithm

$$\begin{aligned} u_1(t,x(\cdot)) &= -\sum_{k=0}^{1} [\beta^*(A_k+\mu I_k)^n x(t-\tau_k(t)) + d_1 x(t-\tau_k(t))] \\ u_i(t,x(\cdot)) &= \sum_{k=0}^{1} d_i x(t-\tau_k(t));\ i=2,3,\ldots,m \end{aligned} \tag{2}$$

then, on substituting (2) in (1) yields

$$\begin{aligned} \dot{x}(t) &= \sum_{k=0}^{1} [A_k - q_1\beta^*(A_k+\mu I_k)^n]\, x(t-\tau_k(t)) \\ &+ \sum_{k=0}^{1} [B_k(t) - Q(Q^*Q)^{-1} Q^* B_k(t)]\, x(t-\tau_k(t) \end{aligned} \tag{3}$$

where $D^* = (d_1^*, \ldots, d_m^*) \triangleq -(Q^*Q)^{-1} Q^* B$, $\mu > 0$ and $\beta > 0$ are arbitrary constants. * denotes the transpose of a matrix. Then eqn. (3) is rewritten as

$$\dot{x}(t) = \sum_{k=0}^{1} [\tilde{A}_k + \tilde{B}_k(t)]\, x(t-\tau_k(t)) \tag{4}$$

where matrices A_k and $B_k(t)$ are defined as

$$\tilde{A}_k = A_k - q_1\beta^*(A_k + \mu I_k)^n \tag{5}$$

and

$$\tilde{B}_k(t) = B_k(t) - Q(Q^*Q)^{-1}Q^* B_k(t) \tag{6}$$

Next we construct Cauchy matrix for the system

$$\dot{x}(t) = \sum_{k=0}^{1} \tilde{A}_k x(t-\tau_k(t)) \tag{7}$$

Cauchy matrix C(t,s) of the system (7) is known to satisfy

$$\begin{aligned} \frac{\partial C(t,s)}{\partial s} &= \tilde{A}_0 C(t,s) + \tilde{A}_1 C(t-\tau_1(t),s), s < t \\ C(t,t) &= I\ ,\ C(t,s) = 0\ ,\ s > t \end{aligned} \tag{8}$$

The physical constraint of the system imposes limitation on the growth rate of the delay $\tau_1(t)$, namely, it should not be greater than the growth rate of the natural course of time. We will assume that $\dot{\tau}_1(t) < 1$. The function $\nu(t)$ is defined as

$$\nu(t) \triangleq t-\tau_1(t) \tag{9}$$

which increases monotonically with $t \geq t_0$ and has the inverse function

$$\rho(t) \in C^1 \; [t_0-\tau_1(t_0),\infty) \tag{10}$$

It has been established that the Cauchy matrix defined by (8) satisfies also the adjoint system

$$\begin{aligned} \frac{\partial C(t,s)}{\partial s} &= -C(t,s)\tilde{A}_0 - C(t,\rho(s))\tilde{A}_1\dot{\rho}(s), \; s < t \\ C(t,t) &= I, \; C(t,s)=0, \; s > t \end{aligned} \tag{11}$$

which is an analogue of semigroup equality of the Cauchy matrix $\Phi(t,s)$ of the system

$$\dot{x}(t) = \tilde{A}_0 x(t) \tag{12}$$

Let $\Phi(t,s)$ be the Cauchy matrix of (12), then we define the Cauchy matrix for (7) as

$$C_i(t,s) = \begin{cases} C_1(t,s) = \Phi(t,s), \; s \in [\nu(t),t] \\ C_i(t,s), \; i=2,\ldots,K(T) \; ; \; for \; s \in [\nu_i(t), \; \nu_{i-1}(t)] \end{cases} \tag{13}$$

where $C_i(t,s)$, $i = 2,\ldots,K(T)$ are solutions of the equations

$$\frac{\partial C_i(t,s)}{\partial s} = -C_i(t,s)\tilde{A}_0 - C_{i-1}(t,\rho(s))\tilde{A}_1\dot{\rho}(s) \tag{14}$$

for $s \; \varepsilon \; [\nu_i(t), \nu_{i-1}(t)]$ and

$$C_i(t,s) = 0, \; s \notin [\nu_i(t), \; \nu_{i-1}(t)] \tag{15}$$

The solution of (14) can be written as

$$C_i(t,s) = C_{i-1}(t,\nu_{i-1}(t))C_1(\nu_{i-1}(t),s) + \int_s^{\nu_{i-1}(t)} C_{i-1}(t,\rho(y))\tilde{A}_1\,\dot{\rho}(y)C_1(y,s)dy \tag{16}$$

for $s \; \varepsilon \; [\nu_i(t), \nu_{i-1}(t)]$, $i = 2,\ldots,K(t)$; $\nu_i(t) \triangleq \nu\,(\nu_{i-1}(t))$, $i=1,2,\ldots$; $\nu_0(t)=t$. We now state the following theorem.

Theorem 1: Let conditions $\tau_1(t) \geq 0$, $\dot{\tau}_1(t) < 1$, $\tau_1(\nu_i(T)) > 0$, for $i=1,\ldots,K(T)$ hold and in addition if $\Phi(T,s)$ is the Cauchy matrix of system (12) for $s \; \varepsilon \; [\nu(T),T]$, then Cauchy matrix $C(T,s)$, $s \; \varepsilon \; [0,T]$ of the system (7) has the representation

$$C(T,s) = \sum_{i=1}^{K(t)} C_i(T,s) \tag{17}$$

where $C_i(t,s)$, defined by expression (13) such that

$$|| C(t,s) || \le M e^{-\mu(t-s)}; \; M > 1 \tag{18}$$

for some $\mu>0$.

Example:

Before proving the theorem, consider the following example, whose Cauchy matrix C(t,s) has been constructed

$$\begin{aligned} \dot{x}_1(t) &= x_1(t) + \alpha x_2(t) + x_1(t-1) \\ \dot{x}_2(t) &= x_1(t) + x_2(t) + x_1(t-1) + 2x_2(t-1) + u(t) \end{aligned} \tag{19}$$

System (19) is rewritten in the form (4) as

$$\dot{x}(t) = [A_0 + B_0(t)]\, x(t) + [A_1 + B_1(t)]\, x(t-\tau(t)) + qu(t) \tag{20}$$

such that

$$\begin{aligned} A_0 &= \begin{pmatrix} -1 & \alpha \\ 1 & 1 \end{pmatrix}, \; B_0 = 0 \\ A_1 &= \begin{pmatrix} 1 & 0 \\ 1 & 2 \end{pmatrix}, \; B_1 = 0, \; q = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \end{aligned} \tag{21}$$

We specify in eqn. (2), $\mu = 1$, and find by the indicated algorithm that β and u(t) are defined by

$$\begin{aligned} \beta^* &= (\alpha^{-1} \; 0) \\ u(t) &= -\sum_{k=0}^{1} \beta^*(A_k + \mu\, I_k)^2\, x(t-\tau_k) \end{aligned} \tag{22}$$

Since, $\tau_0 = 0$, $\tau_1 = 1$, then from eqn. (22) we have

$$u(t) = -x_1(t) - 2x_2(t) - x_1(t-1) - 2x_2(t-1) \tag{23}$$

Therefore, the control synthesized according to (2) is given by (23) when substituted in (19) yields

$$\begin{aligned} \dot{x}_1(t) &= -x_1(t) + \alpha x_2(t) + x_1(t-1) \\ \dot{x}_2(t) &= -x_2(t) + 2x_2(t-1) \end{aligned} \tag{24}$$

We rewrite the system (24) in the form

$$\dot{x}(t) = \begin{pmatrix} -1 & \alpha \\ 0 & -1 \end{pmatrix} x(t) + \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix} x(t-1) \tag{25}$$

In order to construct C(t,s) according to the theorem, we find that the Cauchy matrix C_1 (t,s) = Φ (t,s) = $\Phi(t)\Phi^{-1}(s)$ of the respective system (26) without delay

$$\dot{x}(t) = \begin{pmatrix} -1 & \alpha \\ 0 & -1 \end{pmatrix} x(t) \tag{26}$$

whose fundamental matrix $\Phi(t)$ is given by

$$\Phi(t) = e^{-\mu t} \left[I + (A_0 + \mu I)\, t \right] . \tag{27}$$

For

$$\mu \triangleq 1 , \; A_0 = \begin{pmatrix} -1 & \alpha \\ 0 & -1 \end{pmatrix} \tag{28}$$

we have

$$\Phi(t) = e^{-t} \begin{pmatrix} 1 & \alpha t \\ 0 & 1 \end{pmatrix} \tag{29}$$

Therefore the Cauchy matrix C(t,s) can be written as

$$\begin{aligned} \Phi(t,s) &= e^{-(t-s)} \begin{pmatrix} 1 & \alpha(t-s) \\ 0 & 1 \end{pmatrix} \\ &\triangleq C_1(t,s) \; , \; s \in [t-1, \, t] \end{aligned} \tag{30}$$

From expression (16), $C_2(t,s)$ is obtained as

$$\begin{aligned} C_2(t,s) &= e^{-(t-s)} \begin{pmatrix} 1 & \alpha(t-s) \\ 0 & 1 \end{pmatrix} \\ &+ \int_s^{t-1} e^{-(t-y)} \begin{pmatrix} 1 & \alpha(t-y) \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix} \times e^{-(y-s)} \begin{pmatrix} 1 & \alpha(y-s) \\ 0 & 1 \end{pmatrix} d\,y \end{aligned} \tag{31}$$

For simplicity, choose $\alpha = 1$, then $C_2(t,s)$ for s $\in$ [0, (t-1)] becomes

$$C_2(t,s) = e^{-(t-s)} \begin{pmatrix} t-s & a_{12}(t) \\ 0 & 2(t-s)-1 \end{pmatrix} \tag{32}$$

where

$$a_{12}(t) = \frac{3}{2}\, (t-s)^2 - (t-s) + \frac{3}{2}$$

Thus, for system (19), we have obtained the exact expression of the Cauchy matrix on the interval [0,t].

Proof of Theorem 1

We claim that the Cauchy matrix $C_1(t,s) = \Phi(t,s)$, $s \varepsilon [\nu(t),t]$, for the system below

$$\dot{x}(t) = \tilde{A}_0 x(t) \tag{33}$$

is given by

$$\Phi(t,s) = \exp[-\mu(t-s)] \sum_{i=0}^{n} \frac{(\tilde{A}_0 + \mu I)^{i-1}}{(i-1)!} (t-s)^{i-1} \tag{34}$$

if the eigenvalues of $\tilde{A}_0$ are

$$\Delta_i(\tilde{A}_0) = \mu , \ (i=1,...,n) \tag{35}$$

In order to show this result, we construct a similarity transformation matrix T. Let T be the constant transformation matrix defined by

$$T = (q_1, A_0 q_1, ..., A_0^{n-1} q_1) \tag{36}$$

such that det T ≠ 0. Let β be the solution of the system.

$$T^* \beta = \tilde{e} \tag{37}$$

where $\tilde{e} = Col\,(0,0,...,0,1)$. From the solution β of equation (37) construct the matrix Γ, whose rows are vectors defined as

$$\Gamma = Col\,(\beta^*, \beta^* A_0, ..., \beta^* A_0^{n-1}) \tag{38}$$

It is well known that the matrix Γ has the following properties.

$$(i) \ \ \Gamma q_1 = \tilde{e} \ , \qquad (ii) \ \ \Gamma A_0 \Gamma^{-1} = P_0 \tag{39}$$

where the matrix P_0 is given by

$$P_0 = \begin{pmatrix} 0 & 1 & 0 & . & . & . & 0 \\ 0 & 0 & 1 & . & . & . & 0 \\ . & . & . & . & . & . & . \\ 0 & 0 & 0 & . & . & . & 1 \\ p_1 & p_2 & p_3 & . & . & . & p_n \end{pmatrix} \tag{40}$$

Here p_i, (i=1,...,n) are the coefficients of the characteristic polynomial $\Delta(\tilde{A}_0)$, written in the form $\lambda^n - \sum_{i=1}^{n} p_i \lambda^{(i-1)}$. Let γ_i be the coefficients of the polynomial

$$(\lambda+\mu)^n = \lambda^n + \sum_{i=1}^{n} \gamma_i \lambda^{(i-1)} \tag{41}$$

Now we construct the vector C_0 as

$$C_0 = - Col(p_1+\gamma_1,...,p_n+\gamma_n) \tag{42}$$

and define

$$C^* = C_0^* \Gamma \tag{43}$$

Then it follows that $\tilde{A}_0 = A_0 + q_1 C^*$. Next it can be easily shown that $A_0 + q_1 C^*$ is similar to $P_0 + \tilde{e}\, C_0^*$ under the transformation such that

$$\Gamma \left[A_0 + q_1 C^*\right] \Gamma^{-1} = P_0 + \tilde{e}\, C_0^* \tag{44}$$

The characteristic equation of $P_0 + \tilde{e}\, C_0^*$ is given by

$$\Delta_{P_0+\tilde{e}C_0^*}(\lambda) = (\lambda+\mu)^n \tag{45}$$

Therefore $\lambda_i = \mu$, $(i=1,...,n)$ and the transition matrix $C_1(t,s) = \Phi(t,s)$ is given by (34). Next construct the Cauchy matrices $C_i(T,s)$, $i=2,...,K(T)$. If the delay $\tau_1(t)$ satisfies the condition of the theorem, then the number $K(T)$ is finite and hence the number of intervals $[\nu_i(T), \nu_{i-1}(T)]$, $i=0,1,..., K(T)$ are finite. In the framework of the condition of the theorem, we can construct the Cauchy matrix of the system (7) by means of specific formulas (16) and (34). The kernel in the integral of (16) also has an exponential bound. Now consider equation (4), whose solution through ϕ at $t=t_0$ (i.e. , $x(t_0-s)=\phi(-s)$) is given by

$$x_t = y(t,\phi,t_0) + \int_{t_0}^{t} C(t,s) \sum_{k=0}^{1} \tilde{B}_k(\tau) x_s \, d\tau \tag{46}$$

where $y(t,\phi,t_0)$ is a solution of the linear system (7) through ϕ at $t=t_0$ and $C(t,s)$ is a Cauchy matrix of (7) such that

$$C(s,s) = I \; ; \quad C(s+\theta,s) = 0 \quad for \;\; \theta<0 \tag{47}$$

Estimating (46) with respect to norm yields

$$|x_t| \leq Me^{-\mu t}|\phi| + \int_0^t Me^{-\mu(t-s)} h|x_s|ds \tag{48}$$

where let $|\sum_{k=0}^{1} B_k(t)| \leq h$. Applying Gronwall's inequality to $|x_t|e^{\mu t}$ we obtain

$$|x_t| \leq Me^{-(\mu-Mh)t}|\phi| \tag{49}$$

Thus if μ >Mh, is suitably chosen, then we obtain the result stated in the theorem.

References

[1] Hale, J.K., Theory of functional differential equations, Springer-Verlag, New York, (1977).

[2] Nakagiri, S., On the fundamental solution of delay-differential equations in Banach spaces, J. Differential Equations 41, 349-369, (1981).

[3] Nakagiri, S., Optimal control of linear retarded systems in Banach spaces, J. Math. Anal. Appl., 120, (1986).

[4] Delfour, M.C., The linear quadratic optimal control problem for hereditary differential systems: Theory and numerical solutions, Appl. Math. Optim., 3, 101-162, (1977).

Stability Analysis of Systems with State Saturation and Parameter Uncertainties*

Derong Liu Anthony N. Michel Kaining Wang
Department of Electrical Engineering
University of Notre Dame
Notre Dame, IN 46556
U. S. A.

***Abstract*—For linear systems with parameter uncertainties and subject to state saturation, we establish results concerning the global asymptotic stability of an equilibrium. Systems of the type considered herein capture two important phenomena commonly encountered in the modeling process: (*i*) system parameter uncertainties (which in the present case are modeled by means of interval matrices), and (*ii*) operation of systems over a wide range (which in the present case is accounted for by state saturation nonlinearities). We demonstrate the applicability of the present results by means of several specific examples.**

I INTRODUCTION

IN this paper, we first investigate the stability properties of discrete-time systems described by

$$x(k+1) = \mathrm{sat}[Ax(k)], \quad k = 0, 1, 2, \cdots \qquad (1)$$

where $x(k) \in R^n$, $A = [a_{ij}] \in R^{n\times n}$, $\mathrm{sat}(x) = [\mathrm{sat}(x_1), \mathrm{sat}(x_2), \cdots, \mathrm{sat}(x_n)]^T$, and

$$\mathrm{sat}(x_i) = \begin{cases} 1, & x_i > 1 \\ x_i, & -1 \le x_i \le 1 \\ -1, & x_i < -1 \end{cases} .$$

We assume that in system (1), the matrix A is known to belong to an interval matrix, *i.e.*, $A \in [A^m, A^M]$. (An *interval matrix* $[A^m, A^M]$ with $A^m = [a^m_{ij}] \in R^{n\times n}$, $A^M = [a^M_{ij}] \in R^{n\times n}$, and $a^m_{ij} \le a^M_{ij}$ for al i and j is defined by $[A^m, A^M] \triangleq \{C = [c_{ij}] \in R^{n\times n}: a^m_{ij} \le c_{ij} \le a^M_{ij}, 1 \le i, j \le n\}$.) We will refe to system (1) as a "system with saturation nonlinearities and parameter uncertainties."

Systems with saturation nonlinearities (and with no parameter uncertainties) have been investigated by many researchers (see, e.g., [2], [4], [7]). Systems with parameter uncertainties characterized by interval matrices have also been widely investigated (see e.g., [1], [3], [5], [6]). The stability analysis of systems with saturation nonlinearities *and* parameter uncertainties do not appear to have been addressed. We intend to investigate such a problem in the present paper.

II MAIN RESULTS

Before presenting our results, we introduce some necessary notation.

A square matrix is said to be *positive definite*, if it is symmetric and if all its eigenvalues are positive. A matrix $H = [h_{ij}] \in R^{n\times n}$ is said to be *diagonally dominant*, if

$$|h_{ii}| \ge \sum_{j=1, j\ne i}^{n} |h_{ij}| \text{ for } i = 1, \cdots, n. \qquad (2$$

For I, J, I' and J' such that

$$\begin{aligned} I \cup J = I' \cup J' = \{1, \cdots, n\} \\ I \cap J = I' \cap J' = \phi \end{aligned} \qquad (3)$$

---

*This work was supported in part by the National Science Foundation under grant ECS 91-07728.

we let $A^{IJ}_{I'J'} = [a^{IJI'J'}_{ij}] \in R^{n\times n}$, where

$$a^{IJI'J'}_{ij} = \begin{cases} a^M_{ij} & \text{if } (i,j) \in I' \times I \cup J' \times J \\ a^m_{ij} & \text{if } (i,j) \in I' \times J \cup J' \times I \end{cases}. \quad (4)$$

We will utilize the following definition in our results.

Definition 1 An interval matrix $[A^m, A^M] \in R^{n\times n}$ is said to possess *Property* $\mathcal{R}$ if there exists a positive definite matrix $H \in R^{n\times n}$ which satisfies the diagonal dominance condition (2) such that

$$Q = H - (A^{IJ}_{I'J'})^T H A^{IJ}_{I'J'} \quad (5)$$

is positive definite for all I, J, I' and J' satisfying (3), where $A^{IJ}_{I'J'}$ is defined in (4). ∎

We are now in a position to establish the next theorem.

Theorem 1 The equilibrium $x_e = 0$ of system (1) is *globally asymptotically stable* if the interval matrix $[A^m, A^M]$ possesses Property $\mathcal{R}$. ∎

Our next result utilizes the following definition.

Definition 2 An interval matrix $[A^m, A^M] \in R^{n\times n}$ is said to possess *Property* $\mathcal{T}$ if there exists a positive definite matrix $H \in R^{n\times n}$ which satisfies the diagonal dominance condition (2) such that

$$Q = H - A_0^T H A_0 \quad (6)$$

is positive definite, where $A_0 \triangleq \frac{1}{2}(A^m + A^M)$, and if

$$\frac{1}{2}\alpha(A^M - A^m) < \sqrt{(\alpha(A_0))^2 + \frac{\lambda_m(Q)}{\|H\|_\infty}} - \alpha(A_0) \quad (7)$$

where $\alpha(\cdot)$ is defined as

$$\alpha(W) = \max\{\|W\|_1, \ \|W\|_\infty\} \text{ for } W \in R^{n\times n} \quad (8)$$

and $\lambda_m(Q)$ is the smallest eigenvalue of Q. ∎

Theorem 2 The equilibrium $x_e = 0$ of system (1) is *globally asymptotically stable* if the interval matrix $[A^m, A^M]$ possesses Property $\mathcal{T}$. ∎

Remark 1 Theorem 1 is in general less restrictive than Theorem 2; however, Theorem 2 is considerably easier to apply than Theorem 1, since Theorem 2 involves less computational complexity. If in system (1), $A^m = A^M = \overline{A}$, *i.e.*, the parameters of the system are known exactly, then, Theorem 1 and Theorem 2 reduce to the same result (see [2]), in which case we have $A^{IJ}_{I'J'} = A_0 = \overline{A}$. ∎

REFERENCES

[1] H. Chapellat, M. Dahleh, and S. P. Bhattacharyya, "On robust nonlinear stability of interval control systems," *IEEE Transactions on Automatic Control*, vol. 36, pp. 59–67 (1991)

[2] Derong Liu, A. N. Michel, "Asymptotic stability of discrete-time systems with saturation nonlinearities with applications to digital filters," *IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Applications*, vol. 39, pp. 798–807 (1992)

[3] M. Mansour, "Robust stability of interval matrices," *Proceedings of the 28th IEEE Conference on Decision and Control*, Tampa, Florida, pp. 46–51 (1989)

[4] A. N. Michel, Jie Si, Gune Yen, "Analysis and synthesis of a class of discrete-time neural networks described on hypercubes," *IEEE Transactions on Neural Networks*, vol. 2, pp. 32–46 (1991)

[5] A. N. Michel, Kaining Wang, "Robust stability: Perturbed systems with perturbed equilibria," *Systems & Control Letters*, vol. 20, pp. 155–162 (1993)

[6] T. Mori, H. Kokame, "Convergence property of interval matrices and interval polynomials," *Int. J. Control*, vol. 45, pp. 481–484 (1987)

[7] I. W. Sandberg, "A theorem concerning limit cycles in digital filters," *Proceedings of the 7th Annual Allerton Conference on Circuit and System Theory*, University of Illinois at Urbana-Champaign, Urbana, IL, pp. 63–68 (1969)

On Stability of Time-Delay Systems

G. M. Schoen and H. P. Geering
Measurement and Control Laboratory
Swiss Federal Institute of Technology (ETH)
ETH-Zentrum, CH-8092 Zurich, Switzerland

Many engineering problems lead to an investigation of time-delay systems. In reality it is difficult to estimate the value of the delay, especially if it is time- and/or state-dependent. It is therefore useful to derive stability criteria which do not require the information concerning delay. For the system $\dot{x}(t) = A_0 x(t) + A_1 x(t-\tau)$; $A_0, A_1 \in \boldsymbol{R}^{n \times n}$ with a constant delay [6] gave the following delay independent stability criteria $\mu(A_0) + \|A_1\| < 0$, where $\|A_1\| = \sqrt{\lambda_{max}(A_1^T A_1)}$ and $\mu(A_0) = 0.5 \cdot \lambda_{max}(A_0 + A_0^T)$. Several authors, e.g. [8], indicate that this stability condition is also valid if the delay is time-dependent. The statements above can be generalized further.

Theorem 1: The system $\dot{x}(t) = A_0 x(t) + \sum_{i=1}^{k} A_i x(t - \tau_i(x(t), t))$, with delays satisfying the inequalities $0 \le \tau_i(x(t), t) \le \tau_{max} < \infty$ and $0 \le \dot{\tau}_i(x(t), t) < 1$, is asymptotically stable independent of delay (a.s.i.d.), if the inequality $\mu(A_0) + \sum_{i=1}^{k} \|A_i\| < 0$ is valid.

Proof: Theorem 1 can be derived in two ways, either using the comparison method or the Razumikhin stability theorem [3]. For briefness we omit the proof.

The Razumikhin theorem is used again to investigate the stability of large-scale time-delay systems. In [5], [6], [9], and [10] the stability conditions are derived only for constant delays. The large-scale system considered here consists of N interconnected subsystems S_i, i = 1, 2, ..., N. Each subsystem is described by S_i: $\dot{X}_i(t) = A_i X_i(t) + \sum_{j=1}^{N} A_{ij} X_j(t - \tau_{ij})$, where $X_i \in \boldsymbol{R}^{n_i}$ represents the state of the subsystem S_i. All τ_{ij} are bounded continuous time-varying and state-dependent functions satisfying the inequalities $0 \le \tau_{ij}(X_j(t), t) \le \tau_{max}$ and $0 \le \dot{\tau}_{ij}(X_j(t), t) < 1$.

Theorem 2: The system S_i is a.s.i.d., if $\mu(A_i + \frac{1}{2}\sum_{j=1}^{N} A_{ij}^T A_{ij}) + \frac{N}{2} < 0 \qquad i = 1, ..., N$.

Proof: The Lyapunov-Razumikhin function is chosen to be of the class of quadratic forms $V = \sum_{i=1}^{N} V_i = \sum_{i=1}^{N} X_i(t)^T X_i(t)$. Determining the derivative of V yields

$$\sum_{i=1}^{N} \dot{V}_i = \sum_{i=1}^{N} \{X_i(t)^T [A_i^T + A_i] X_i(t) + 2 \sum_{j=1}^{N} X_i(t)^T A_{ij} X_j(t - \tau_{ij})\}. \qquad (1)$$

We now apply the fact that for any two matrices U_1 and U_2 with appropriate dimensions the relation $U_1^T U_2 + U_2^T U_1 \le U_1^T U_1 + U_2^T U_2$ holds. Furthermore, according to [3], the inequality $\sum_{i=1}^{N} V_i(X_i(t)) \ge \sum_{i=1}^{N} V_i(X_i(\theta))$ for all $\theta \in [t - \tau_{max}, t]$ must be satisfied to ensure asymptotic stability. Thus, (1) yields

$$\sum_{i=1}^{N} \dot{V}_i < x(t)^T \mathrm{diag}(M_1, \dots, M_N)\, x(t)\,, \tag{2}$$

where $x^T(t) = [X_1^T(t), \dots, X_N^T(t)]$ and $M_i = A_i^T + A_i + \sum_{j=1}^{N} A_{ij}^T A_{ij} + I \cdot N$. The identity matrix is $I \in \mathbf{R}^{n_i \times n_i}$. Since $\dot{V}_i$ has to be negative-definite, the statement of Theorem 2 follows from inequality (2) □.

Example 1: In [5] and [9] the asymptotic stability of a numerically given large-scale system was proved for a constant delay. Applying Theorem 2 it can be shown that this system is also asymptotically stable if the delay is not constant. In contrast to the methods of [5], [9] no complex Lyapunov equation needs to be solved.

Reduction of the Conservativity of the Stability Criteria: The stability criteria presented above can be remarkably improved by applying an appropriately chosen transformation matrix. Using transformed state vector techniques, it is easy to show that Theorem 1 can be rewritten as $\mu(T^{-1}A_0T) + \sum_{i=1}^{k} \|T^{-1}A_iT\| < 0$, where T is a regular matrix. From Theorem 2 we obtain $\mu(T_i^{-1}A_iT_i + 0.5\sum_{j=1}^{N} (T_i^{-1}A_{ij}T_j)^T T_i^{-1}A_{ij}T_j) + \frac{N}{2} < 0;\, i = 1, 2, \dots, N$. The determination of the matrix T (or T_i, respectively,) which minimizes the left-hand side of the inequality leads to nonlinear conditions, in general. However, simple numerical methods are very effective to find an appropriate matrix T or T_i (e.g., steepest descent method, threshold accepting, simulated annealing).

Example 2: The William-Otto process consists of a chemical reactor, a cooler, a decanter, and a distillation column. In [4] the model of the refining plant is given as a single delay system.

$$\dot{x}(t) = \begin{bmatrix} -4.93 & -1.01 & 0 & 0 \\ -3.2 & -5.3 & -12.8 & 0 \\ 6.4 & 0.347 & -32.5 & -1.04 \\ 0 & 0.833 & 11.0 & -3.96 \end{bmatrix} x(t) + \begin{bmatrix} 1.92 & 0 & 0 & 0 \\ 0 & 1.92 & 0 & 0 \\ 0 & 0 & 1.87 & 0 \\ 0 & 0 & 0 & 0.724 \end{bmatrix} x(t-1) + \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} u(t)\,.$$

We obtain $\mu(A_0) + \|A_1\| = 0.657 \not< 0$. Using $T = \mathrm{diag}[-0.37; 1.30; -0.52; 1.06]$ in the similarity transformation, we obtain $\mu(T^{-1}A_0T) + \|T^{-1}A_1T\| = -0.61 < 0$. This shows that the system is asymptotically stable for a nonconstant delay. Using the same stability criterion it can be proved that the LQ-regulator design method (averaging-method) applied in [4] generates a closed-loop system, which is a.s.i.d. The closed-loop system considered is also a.s.i.d. if the estimation of the delay is wrong, and/or if the delay of the plant is not constant.

The investigation of the stability of other LQ-regulator design methods for delay-systems (e.g. spline-Galerkin method, Legendre-tau method) and the problem of finding sufficient conditions for the stability of time-delay systems with nonlinear time-varying pertubations will be considered in our future work.

Modified D-Decomposition Method: Sufficient stability and instability criteria together with the D-decomposition method [2] can be used to derive necessary and sufficient stability conditions, for time-delay systems with *constant* delays: $\dot{x}(t) = A_0x(t) + \sum_{i=1}^{k} A_ix(t-\tau_i)$. In the usual D-decomposition method the coefficient space is partitioned into different regions by means of hypersurfaces the points of which are characterized by the corresponding characteristic equation: $sI - A_0 - \sum_{i=1}^{k} A_i e^{-s\tau_i} = 0$ having at least one zero on the imaginary axis. The points in the interior of each region correspond to a characteristic equation with the same number of zeros with positive real parts. The number of zeros with positive real parts can only change when a zero passes across the imaginary axis, i.e., when the point in the coefficient space passes across the boundary of the region. Here, the goal is finding all of the regions in the coefficient space where the characteristic equation has no zero with positive real part. These regions κ are regions of asymptotic stability. In order to check how the number of roots with positive real parts changes when crossing the boundary, the differential $d\sigma$ of the real part of the root (which crosses the imaginary axis) is usually determined. The number of roots with positive real parts decreases (increases) if the sign of $d\sigma$ is negative (positive). However, this procedure is not always well-suited for singling out asymptotically stable regions because, in general, there is an infinite number of boundaries which are given in parametric form. In order to improve the search of the regions κ, Theorem 1 and the instability criterion of [1] are used. A result of this method is the following theorem.

***Theorem** 3*: The system $\dot{x}(t) = a_0 x(t) + a_1 x(t-h) + a_2 x(t-2h)$ with $|a_2| < \pi/(2h)$ is asymptotically stable if and only if the following three conditions hold for some $y \in [0, \pi/h)$

$$a_0 + a_1 + a_2 < 0 \, ,$$

$$a_0 = \frac{y\cos(yh)}{\sin(yh)} + a_2 \, ,$$

$$a_1 > -\frac{y}{\sin(yh)} - 2a_2\cos(yh) \; .$$

Proof: The proof can be found in [7]. However, the general ideas of the method described above were not presented in [7]. We intend to apply this procedure for more general systems.

REFERENCES

[1] BUSLOWICZ, M., 1985, *Found. Contr. Engin.*, **10**, 11.

[2] EL' SGOL'TS, L., E., 1966, *Introduction to the Theory of Differential Equations with Deviating Arguments,* San Francisco: Holden-Day

[3] HALE, J., 1977, *Theory of Functional Differential Equations*, N.Y.: Springer-Verlag.

[4] HAMMARASTROEM, L., G., GROS, K., S., 1980, *Int. J. Contr.*, **32**, 329.

[5] HMAMED, A., 1986, *Int. J. Systems Sci.*, **17**, 1083.

[6] MORI, T., FUKUMA, N., and KUWAHARA, M., 1981, *Int. J. Contr.*, **34**, 1175.

[7] SCHOEN, G., M., and GEERING, H., P., 1993, *Int. J. Contr.*, **58**, 247.

[8] SU, T., J., and HUANG, C., G., 1992, *IEEE Trans. Autom. Contr.*, **37**, 1656.

[9] WANG, W. J., and SONG, C. C., 1989, *Contr. Theory Adv. Tech.*, **5**, 315.

[10] WANG, W. J., SONG, C. C., and KAO, C. C., 1991, *Int. J. Systems Sci.*, **22**, 209.

NECESSARY AND SUFFICIENT CONDITIONS FOR THE HURWITZ STABILITY OF INTERVAL MATRICES OF ARBITRARY ORDER WITH KNOWN ALGORITHMIC COMPLEXITY FOR THIRD ORDER *

Hui Ye Kaining Wang Anthony N. Michel
Department of Electrical Engineering
University of Notre Dame
Notre Dame, IN 46556, U.S.A.

Abstract

We establish a set of necessary and sufficient conditions for the Hurwitz stability of interval matrices of arbitrary order. We use these conditions to show that for a given real third order interval matrix, there exists a finite subset of the given interval matrix (consisting of at most 1472 matrices) having the property that the given interval matrix is Hurwitz stable if and only if every matrix in the finite subset is Hurwitz stable. A complete description of such a finite subset for any third order interval matrix is given. We demonstrate the applicability of our results by means of two specific examples.

I. Introduction

The stability of interval matrices is of great current interest (see, e.g., [1], [2], [4]–[6], [8]–[12]). In most of these papers, sufficient conditions for stability are established and in some of them necessary conditions are given. Although there are few results which provide necessary and sufficient conditions, these conditions are in general not practical to verify. To the best of the authors' knowledge, there is no known existing method which enables one to verify the Hurwitz stability of arbitrary $n \times n$ real interval matrices for $n \geq 3$ in a finite number of computation steps where the number of computation steps is known a *priori.*

In the present paper, we establish necessary and sufficient conditions for the Hurwitz stability of interval matrices which enable us to ascertain numerically, in a finite number of steps, the Hurwitz stability of most but not all interval matrices of arbitrary order. *More importantly, the present results provide a complete solution to this problem for n=3.* Specifically, our results (Proposition 1 and Theorem 2) enable us to implement an algorithm for determining the Hurwitz stability of an arbitrary 3×3 real interval matrix *with a known upper bound for the number of computation steps.* It is known that for a

---

*This work was supported in part by the National Science Foundation under Grant ECS91-07728.

3×3 real interval matrix, the Hurwitz stability of the $2^9 = 512$ corner matrices of the interval does not imply the Hurwitz stability of the interval matrix (see, e.g., [8]). In the present paper, we identify for any 3×3 real interval matrix a *finite* subset of the interval matrix with the property that the Hurwitz stability of all matrices in this finite subset implies the Hurwitz stability of the entire interval matrix (Theorem 2). We show that for any 3×3 real interval matrix, such a subset consists of at most 1472 matrices in the interval matrix. Furthermore, we present explicit formulas for determining such a finite subset.

In the next section, we provide the notation used throughout this paper. In section 3, we establish necessary and sufficient conditions for the Hurwitz stability of interval matrices of arbitrary order (Theorem 1). In arriving at this result, we need to evaluate the range of the determinants of interval matrices. Therefore, in section 3 we prove that the range of the determinants of any interval matrix is a closed interval and both endpoints of this interval are the determinants of certain corner matrices (Lemma 3). In section 4, we apply Theorem 1 to solve the Hurwitz stability problem completely in the sense discussed above for $n = 3$. In section 5, we present an algorithm which determines the Hurwitz stability of 3×3 interval matrices. We demonstrate the applicability of this algorithm by means of two specific examples.

II. Notation

Let R denote the set of real numbers. For $a \leq b$, the *closed interval* $[a, b]$ is defined to be the set $\{c : a \leq c \leq b\}$ with *endpoints* a and b.

Let R^n denote real n-space. If $x \in R^n$, then $x^T = (x_1, \ldots, x_n)$ denotes the transpose of x. For $x, y \in R^n$, if $x_i \leq y_i$ for $i = 1 \ldots n$, we call the set $\{(w_1, \ldots, w_n)^T : x_i \leq w_i \leq y_i, i = 1 \ldots n\}$ a *hyper-rectangle* determined by x and y. The set of all points in such a hyper-rectangle having $(n - k)$ coordinates as endpoints is called a *k-face* of the hyper-rectangle.

Let $R^{n \times m}$ denote the set of $n \times m$ real matrices. If $B = [b_{ij}]_{n \times m} \in R^{n \times m}$, then B^T denotes the transpose of B. For $A^m = [a_{ij}^m]_{n \times n}$ and $A^M = [a_{ij}^M]_{n \times n}$ satisfying $a_{ij}^m \leq a_{ij}^M$ for all $1 \leq i, j \leq n$, we define the *interval matrix* $[A^m, A^M]$ by $[A^m, A^M] = \{A = [a_{ij}] : a_{ij}^m \leq a_{ij} \leq a_{ij}^M, \ 1 \leq i, j \leq n\}$. The *corner matrices* of this interval consist of the set of all matrices having elements either from A^m or A^M. If for all $A \in [A^m, A^M]$, A is Hurwitz stable (i.e., all eigenvalues of A have negative real parts), we say that $[A^m, A^M]$ is *Hurwitz stable*.

III. Necessary and Sufficient Conditions for Hurwitz Stability of Interval Matrices

For $A \in R^{n \times n}$, consider the Lyapunov equation $PA + A^T P = -I$, where $P = P^T \in R^{n \times n}$ and $I \in R^{n \times n}$ denotes the identity matrix. Since P is symmetric, there are $\frac{n(n+1)}{2}$ variables in the above Lyapunov equations. Let $p_1, \ldots, p_{\frac{n(n+1)}{2}}$ be an enumeration of these variables and define the vector $\tilde{p} = (p_1, p_2, \ldots, p_{\frac{n(n+1)}{2}})^T$. We can rewrite the Lyapunov equation as another linear equation

$$\tilde{A}\tilde{p} = b, \tag{3.1}$$

where $b = (-\frac{1}{2}, 0, \ldots, -\frac{1}{2}, 0, \ldots, -\frac{1}{2})^T$. For given $A \in R^{n \times n}$, the matrix $\tilde{A}$ is unambiguously determined by (3.1).

Using the definition of $\tilde{A}$, we establish the main result of this section.

Theorem 1 For an interval matrix $[A^m, A^M]$, suppose that there exists a matrix $A_0 \in [A^m, A^M]$ which is Hurwitz stable. Then $[A^m, A^M]$ is Hurwitz stable if and only if

(i) the determinants of all corner matrices of $[A^m, A^M]$ have identical sign, and

(ii) for all $A \in [A^m, A^M]$, $f(A) \triangleq \prod_{i<j}(\lambda_i + \lambda_j) \neq 0$, where λ_i, $i = 1 \ldots n$, are the eigenvalues of A.

We will present the proof of this result in several steps. In doing so, we need to establish the following lemmas.

Lemma 1 If $A \in R^{n \times n}$ has one of its eigenvalues on the imaginary axis, then $det(\tilde{A}) = 0$.

Proof. First, we note that $det(\tilde{A}) = \lambda_1 \ldots \lambda_n \prod_{i<j}(\lambda_i + \lambda_j)$ (see, e.g., [3]), where $\lambda_1, \ldots, \lambda_n$ are the eigenvalues of A. Suppose that λ_i is an eigenvalue of A on the imaginary axis. If $\lambda_i = 0$, we certainly have $det(\tilde{A}) = 0$. If $\lambda_i \neq 0$, there exists another eigenvalue $\lambda_j = -\lambda_i$, since the nonreal eigenvalues always appear in conjugate pairs. This leads to $\prod_{i<j}(\lambda_i + \lambda_j) = 0$ which also implies that $det(\tilde{A}) = 0$. □

By using Lemma 1, we establish the following result.

Lemma 2 For an interval matrix $[A^m, A^M]$, suppose that there exists an $A_0 \in [A^m, A^M]$ which is Hurwitz stable. Then the interval matrix $[A^m, A^M]$ is Hurwitz stable if and only if for all $A \in [A^m, A^M]$, $det(\tilde{A}) \neq 0$.

Proof. (Necessity) If there exists an $A \in [A^m, A^M]$ such that for the matrix $\tilde{A}$ generated by A we have $det(\tilde{A}) = 0$, then either $\lambda_k = 0$ for some $1 \leq k \leq n$, or $\lambda_i = -\lambda_j$ for some $1 \leq i, j \leq n$, where the $\lambda_i, \lambda_j, \lambda_k$ denote eigenvalues of A. Both conditions imply that A is not Hurwitz stable. This proves the necessity.

(Sufficiency) Assume that $[A^m, A^M]$ is not Hurwitz stable, i.e., there exists an $A_1 \in [A^m, A^M]$ which is not Hurwitz stable. This means that A_1 has an eigenvalue with nonnegative real part. Also, assume that A_0 is Hurwitz stable, i.e., the real parts of all eigenvalues of A_0 are negative. By continuity, there must exist a matrix A_2 in the interval $[A^m, A^M]$ such that A_2 has at least one eigenvalue on the imaginary axis. By Lemma 1, we have $det(\tilde{A}_2) = 0$. This contradicts the assumption that $det(\tilde{A}) \neq 0$ for all $A \in [A^m, A^M]$. Therefore, $[A^m, A^M]$ is Hurwitz stable. □

Lemma 2 tells us that in order to determine the Hurwitz stability of an interval matrix $[A^m, A^M]$, it suffices to know whether the range of $det(\tilde{A})$ includes zero as A varies over the interval $[A^m, A^M]$, assuming that there exists at least one matrix in this interval which is Hurwitz stable. Since $det(A) = \lambda_1 \ldots \lambda_n$, we know that $det(\tilde{A}) = det(A) \prod_{i<j}(\lambda_i + \lambda_j)$. Thus, $det(\tilde{A})$ can be zero if either $det(A)$ is zero or $\prod_{i<j}(\lambda_i + \lambda_j)$ is zero. In the next lemma, we will study the range of determinants of interval matrices.

Lemma 3 Let $[A^m, A^M]$ be an interval matrix. The range of $det(A)$ as A varies over $[A^m, A^M]$ is a closed real interval, and the endpoints of this interval are the determinants of some corner matrices of $[A^m, A^M]$.

Proof. The determinant of a real $n \times n$ matrix is a continuous function from $R^{n\times n}$ to R, and $[A^m, A^M]$ is a compact and connected subset in $R^{n\times n}$. Since continuous functions (of $R^{n\times n}$ to R) preserve compactness and connectedness of sets (see e.g., [7]), it follows that the range of $det(A)$ as A varies over $[A^m, A^M]$ is also a compact and connected subset in R (which has to be a closed interval in R).

A determinant of a matrix is a function of all elements of the matrix. For each a_{ij}, $1 \leq i, j \leq n$, we have

$$\frac{\partial det(A)}{\partial a_{ij}} = det(A_{ij}) \tag{3.2}$$

where A_{ij} is the cofactor of a_{ij}. Since $det(A_{ij})$ is independent of a_{ij}, it follows that $det(A)$ is a linear function of each a_{ij} with all other elements fixed. Since a one variable linear function on a closed interval attains its extreme values at the endpoints, it follows that the determinant function assumes its extreme values at corner matrices. □

As a consequence of Lemma 3, we can evaluate the range of the determinants of all matrices in a given interval matrix precisely. In doing so, we evaluate the determinants of all corner matrices, and choose the largest and smallest values as the endpoints of a closed interval. This interval is precisely the range of $det(A)$ as A varies over $[A^m, A^M]$.

The proof of Theorem 1 follows by combining Lemmas 2 and 3.

Remarks 1) An equivalent statement of Lemma 2, but in a very different form has been reported in [12].

2) Theorem 1 will enable us to solve completely the Hurwitz stability problem of 3×3 interval matrices in the following sense: for a given 3×3 interval matrix we identify explicitly a *finite* subset of the given interval matrix having the property that the Hurwitz stability of all matrices in this finite subset implies the Hurwitz stability of the given interval matrix. The particular forms of Lemma 2 and 3 enable us to generate an algorithm for 3×3 interval matrices which has the following attributes: the algorithm will determine whether or not a 3×3 interval matrix is Hurwitz stable using at most 64 cycles. □

Lemma 3 enables us to determine the range of the determinants of all matrices in an interval matrix. However, the determination of the range of $det(\tilde{A})$ as A varies over $[A^m, A^M]$ is more complicated than the determination of the range of $det(A)$ as A varies over $[A^m, A^M]$, for $det(\tilde{A})$ is no longer a linear function of a_{ij}. Although the range of $det(\tilde{A})$ is a closed interval, it is difficult to determine the values of the endpoints of this closed interval for the general case. We solve this problem when $n = 3$ in the next section.

Generally, when A varies over an interval matrix $[A^m, A^M]$, the set $\{\tilde{A} | A \in [A^m, A^M]\}$ does not constitute an interval matrix; however this set is always a subset of some interval matrix, say $[\hat{A}^m, \hat{A}^M]$. By Lemma 3, we can determine the range of $det(\hat{A})$ as $\hat{A}$ varies over $[\hat{A}^m, \hat{A}^M]$. This range which contains the actual range of $det(\tilde{A})$ will be called a *rough estimate* of the range of $det(\tilde{A})$. Therefore, for any interval matrix $[A^m, A^M]$, we can determine a rough estimate of the range of $det(\tilde{A})$. In order to determine whether the actual range of $det(\tilde{A})$ includes zero or not, we can apply classical numerical methods of interval analysis. That is, we can divide $[A^m, A^M]$ into a finite number of subintervals, and obtain a rough estimate for the range of the determinants of the matrices generated by each subinterval matrix. The union of all the rough estimates will cover the actual range of $det(\tilde{A})$ as A varies over $[A^m, A^M]$. This union will converge to the actual range as the radii of the subintervals approach zero. Accordingly, for the general case we are able to determine the Hurwitz stability of an interval matrix in a finite number of computation steps for all cases where zero is not an endpoint of the range of $det(\tilde{A})$. Therefore, our method applies to *most* interval matrices; however for the reasons enumerated in the following (pertaining to complexity), our method does not solve the interval matrix Hurwitz stability problem completley: (a) We do not know an upper bound for the number of computation steps required in our algorithm, and (b) there are no guarantees that our algorithm will terminate in a finite number of steps if zero is an endpoint of the range of $det(\tilde{A})$ as A varies over $[A^m, A^M]$. In the next section, we remove these deficiencies for the case $n = 3$ (The case $n = 2$ has been solved some time ago. (see, e.g. [10])).

IV. A Complete Solution For 3×3 Interval Matrices

In order to arrive at the main results of the present section (Proposition 1 and Theorem 2), we will require two preliminary results (Corollary 1 and Lemma 4).

Let $\lambda_i, i = 1 \dots n$ denote the eigenvalues of a matrix $A \in R^{n\times n}$. For $n = 3$, we have $det(\tilde{A}) = \lambda_1\lambda_2\lambda_3(\lambda_1+\lambda_2)(\lambda_1+\lambda_3)(\lambda_2+\lambda_3)$. It is easily shown that if $A \in R^{3\times 3}$ is Hurwitz stable, then $\lambda_1\lambda_2\lambda_3 < 0$ and $(\lambda_1+\lambda_2)(\lambda_1+\lambda_3)(\lambda_2+\lambda_3) < 0$. As a consequence of Lemma 2, we therefore have the following result.

Corollary 1 Assume that $[A^m, A^M]$ is a 3×3 real interval matrix and there exists at least one matrix in this interval matrix which is Hurwitz stable. Then $[A^m, A^M]$ is Hurwitz stable if and only if (i) $\lambda_1\lambda_2\lambda_3 < 0$ and (ii) $(\lambda_1 + \lambda_2)(\lambda_1 + \lambda_3)(\lambda_2 + \lambda_3) < 0$ for all $A \in [A^m, A^M]$, where λ_1, λ_2, and λ_3 are the eigenvalues of A.

Proof. The necessity is obvious. For the sufficiency, observe that if (i) and (ii) are true, then $det(\tilde{A}) \neq 0$. By Lemma 2, $[A^m, A^M]$ is stable. □

We know that $det(\tilde{A}) = \lambda_1\lambda_2\lambda_3(\lambda_1 + \lambda_2)(\lambda_1 + \lambda_3)(\lambda_2 + \lambda_3) = det(A)(\lambda_1 + \lambda_2)(\lambda_1 + \lambda_3)(\lambda_2 + \lambda_3)$. Let f be the function of A which was defined in Theorem 1, i.e., $f(A) = (\lambda_1 + \lambda_2)(\lambda_1 + \lambda_3)(\lambda_2 + \lambda_3)$. For $A = [a_{ij}]_{3\times 3}$, define $s = \lambda_1 + \lambda_2 + \lambda_3$. Then

$$\begin{aligned} f(A) &= (s - \lambda_1)(s - \lambda_2)(s - \lambda_3) \\ &= s^3 - (\lambda_1 + \lambda_2 + \lambda_3)s^2 + (\lambda_1\lambda_2 + \lambda_1\lambda_3 + \lambda_2\lambda_3)s - \lambda_1\lambda_2\lambda_3 \\ &= (\lambda_1\lambda_2 + \lambda_1\lambda_3 + \lambda_2\lambda_3)(\lambda_1 + \lambda_2 + \lambda_3) - \lambda_1\lambda_2\lambda_3 \\ &= \left(det\left(\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}\right) + det\left(\begin{bmatrix} a_{11} & a_{13} \\ a_{31} & a_{33} \end{bmatrix}\right) + det\left(\begin{bmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{bmatrix}\right)\right) \\ &\quad \cdot(a_{11} + a_{22} + a_{33}) - det(A). \end{aligned} \tag{4.1}$$

Since $det(\tilde{A}) = det(A)f(A)$, it follows from Corollary 1 that in order to verify the Hurwitz stability of $[A^m, A^M]$, we first need to determine whether $det(A)$ is negative for all $A \in [A^m, A^M]$ (We are able to do this by Lemma 3). If this is not the case, then the interval matrix is surely not Hurwitz stable. If it is the case, we only need to verify if the maximal value of $f(A)$ is less than zero. The following result is helpful in evaluating the range of $f(A)$.

Lemma 4 Let $f(A)$ be the real valued function defined by (4.1) on $[A^m, A^M] \subset R^{3\times 3}$ and let $A^m = [a_{ij}^m]$ and $A^M = [a_{ij}^M]$, $1 \leq i,j \leq 3$. Then, there exist $A_1 = [a_{ij}^1]$ and $A_2 = [a_{ij}^2]$ in $[A^m, A^M]$, where for any $i \neq j$ a_{ij}^1 is either a_{ij}^m or a_{ij}^M and a_{ij}^2 is either a_{ij}^m or a_{ij}^M, such that f attains its maximal value at A_1 and minimal value at A_2.

Proof. We observe that $f([a_{ij}]_{3\times 3})$ is a linear function of its individual nondiagonal variables with remaining variables fixed. Therefore, the conclusion of the lemma follows from the fact that a one variable linear function on a closed interval attains its extreme values at the endpoints of the closed interval. □

In view of this lemma, to determine the extreme values of $f(A)$, we only need to consider endpoints of the nondiagonal elements of A. In doing so, we fix these elements to their endpoints, so that f is only a function of a_{11},a_{22} and a_{33}. This results in a total of $2^6 = 64$ functions. We denote these functions by $f_i(a_{11}, a_{22}, a_{33})$, $1 \leq i \leq 64$, which are defined on $H \triangleq [a_{11}^m, a_{11}^M] \times [a_{22}^m, a_{22}^M] \times [a_{33}^m, a_{33}^M]$. In order to determine the extreme values of $f(A)$ as A varies over $[A^m, A^M]$, it suffices to determine the extreme values of $f_i(a_{11}, a_{22}, a_{33})$ as (a_{11}, a_{22}, a_{33}) varies over H, for all $1 \leq i \leq 64$. This is true since $max\{f(A) : A \in [A^m, A^M]\} = max\{f_i(a_{11}, a_{22}, a_{33}) : (a_{11}, a_{22}, a_{33}) \in H; 1 \leq i \leq 64\}$ and since similar relations also hold for minimal values.

From basic Calculus, for each fixed i, $1 \leq i \leq 64$, the extreme values of $f_i(a_{11}, a_{22}, a_{33})$ are attained at points contained in the union of the following four sets S_0, S_1, S_2 and S_3: S_0 is the set of all vertices of H, i.e.,

$$S_0 \triangleq \{(a_{11}, a_{22}, a_{33}) : a_{jj} = a_{jj}^m \ or \ a_{jj} = a_{jj}^M, j = 1,2,3\}; \tag{4.2}$$

S_1 is the set of all critical points on the 1-face of H, i.e.,

$$\begin{aligned} S_1 \triangleq \ & \cup_{j=1}^3 \{(a_{11}, a_{22}, a_{33}) \in H : a_{kk} = a_{kk}^m \ or \ a_{kk} = a_{kk}^M, \\ & for \ 1 \leq j \neq k \leq 3 \ and \ \frac{\partial f_i}{\partial a_{jj}}(a_{11}, a_{22}, a_{33}) = 0\}; \end{aligned} \tag{4.3}$$

S_2 is the set of all critical points on the 2-face of H, i.e.,

$$\begin{aligned} S_2 \triangleq \ & \cup_{j=1}^3 \{(a_{11}, a_{22}, a_{33}) \in H : a_{jj} = a_{jj}^m \ or \ a_{jj} = a_{jj}^M, \\ & and \ \frac{\partial f_i}{\partial a_{kk}}(a_{11}, a_{22}, a_{33}) = 0, \ for \ 1 \leq j \neq k \leq 3\}; \end{aligned} \tag{4.4}$$

and S_3 is the set of all critical points in the interior of H, i.e.,

$$S_3 \triangleq \{(a_{11}, a_{22}, a_{33}) \in H : \frac{\partial f_i}{\partial a_{kk}}(a_{11}, a_{22}, a_{33}) = 0 \ for \ k = 1,2,3\}. \tag{4.5}$$

Therefore, the determination of the extreme values of f_i on S_0, S_1, S_2 and S_3 enables us to determine the extreme values of f_i on H. Therefore, when there exists at least one Hurwitz stable matrix in this interval matrix and the determinants of all corner matrices are negative, the interval matrix is Hurwitz stable if and only if $max\{f_i(a_{11}, a_{22}, a_{33}) : (a_{11}, a_{22}, a_{33}) \in S_0 \cup S_1 \cup S_2 \cup S_3, 1 \leq i \leq 64\} < 0$. In [1] it is shown that any $n \times n$ interval matrix is Hurwitz stable if and only if all the matrices on the $2n - 4$ faces of the interval matrix are Hurwitz stable, where the interval matrix is viewed as an n^2

dimensional hyper-rectangle. Therefore, when there exists at least one Hurwitz stable matrix in this interval matrix and the determinants of all corner matrices are negative, it follows that the interval matrix is Hurwitz stable if and only if $max\{f_i(a_{11}, a_{22}, a_{33}) : (a_{11}, a_{22}, a_{33}) \in S_0 \cup S_1 \cup S_2, 1 \leq i \leq 64\} < 0$. Hence, we do not need to consider S_3.

Since there are a finite number of points in S_0, it is straightforward to determine the extreme values on S_0. To determine the sets $S_i, i = 1, 2$, we utilize partial derivatives of f_i. Without loss of generality, we demonstrate the evaluation of $\frac{\partial f_i}{\partial a_{11}}$:

$$\begin{aligned}\frac{\partial f_i}{\partial a_{11}} &= \left(det\left(\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}\right) + det\left(\begin{bmatrix} a_{11} & a_{13} \\ a_{31} & a_{33} \end{bmatrix}\right) + det\left(\begin{bmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{bmatrix}\right)\right) \\ &\quad +(a_{22}+a_{33})(a_{11}+a_{22}+a_{33}) - det\left(\begin{bmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{bmatrix}\right) \\ &= (a_{22}+a_{33})^2 + 2a_{11}(a_{22}+a_{33}) - (a_{12}a_{21} + a_{13}a_{31}). \end{aligned} \tag{4.6}$$

Then, $\frac{\partial f_i}{\partial a_{11}} = 0$ implies

$$(a_{11}+a_{22}+a_{33})^2 = a_{11}^2 + a_{12}a_{21} + a_{13}a_{31}.$$

Recall that in Theorem 1 and Corollary 1, we assume that there exists a Hurwitz stable matrix $A_0 \in [A^m, A^M]$. To ensure that this is true, we verify that A^M is Hurwitz stable (for otherwise, the interval matrix is certainly not Hurwitz stable by definition).

In view of the above, we assume henceforth that A^M is Hurwitz stable. Then $trace(A^M) < 0$, and $trace(A) \leq trace(A^M) < 0$ for all $A \in [A^m, A^M]$. We therefore have

$$a_{11}+a_{22}+a_{33} = -\sqrt{a_{11}^2 + a_{12}a_{21} + a_{13}a_{31}}. \tag{4.7}$$

In a similar manner we can show that , $\frac{\partial f_i}{\partial a_{22}} = 0$ implies

$$a_{11}+a_{22}+a_{33} = -\sqrt{a_{22}^2 + a_{12}a_{21} + a_{23}a_{32}}, \tag{4.8}$$

and $\frac{\partial f_i}{\partial a_{33}} = 0$ implies

$$a_{11}+a_{22}+a_{33} = -\sqrt{a_{33}^2 + a_{13}a_{31} + a_{23}a_{32}}. \tag{4.9}$$

For the set S_1, only one of the elements a_{11}, a_{22}, a_{33} is to be determined while the other two elements are endpoints. Without loss of generality, we consider the case when only a_{11} is to be determined. From (4.6), $\frac{\partial f_i}{\partial a_{11}} = 0$ implies

$$a_{11} = \frac{(a_{12}a_{21}+a_{13}a_{31}) - (a_{22}+a_{33})^2}{a_{22}+a_{33}} \tag{4.10}$$

if $a_{22}+a_{33} \neq 0$. For the case $a_{22}+a_{33} = 0$, there are two possibilities. The first is that there is no solution for $\frac{\partial f_i}{\partial a_{11}} = 0$ if the denominator of (4.10) is nonzero, and the second is that $\frac{\partial f_i}{\partial a_{11}} \equiv 0$ if the denominator of (4.10) is zero. In the first case, there is no critical point on the 1-face for these special a_{22} and a_{33}, where $a_{22}+a_{33} = 0$; in the second case, we have

$$f_i(a_{11}, a_{22}, a_{33}) = f_i(a_{11}^m, a_{22}, a_{33}) = f_i(a_{11}^M, a_{22}, a_{33}), \tag{4.11}$$

for all $a_{11} \in [a_{11}^m, a_{11}^M]$. Since a_{22} and a_{33} are endpoints, $(a_{11}^m, a_{22}, a_{33})$ and $(a_{11}^M, a_{22}, a_{33})$ are vertices of H. Based on these observations, if we define

$$\begin{aligned}\tilde{S}_1 \;&\triangleq\; \cup_{j=1}^{3}\{(a_{11}, a_{22}, a_{33}) \in H : a_{kk} = a_{kk}^m \textit{ or } a_{kk}^M \textit{ for } 1 \le j \ne k \le 3 \\ &\quad \textit{and } \frac{\partial f_i}{\partial a_{jj}}(a_{11}, a_{22}, a_{33}) = 0, a_{11} + a_{22} + a_{33} - a_{jj} \neq 0\},\end{aligned} \tag{4.12}$$

then $\tilde{S}_1 \subset S_1$ and $max\{f_i(a_{11}, a_{22}, a_{33}) : (a_{11}, a_{22}, a_{33}) \in S_0 \cup S_1 \cup S_2\} = max\{f_i(a_{11}, a_{22}, a_{33}) : (a_{11}, a_{22}, a_{33}) \in S_0 \cup \tilde{S}_1 \cup S_2\}$, and similar relations hold for minimal values. We are able to determine the extreme values of $f_i(a_{11}, a_{22}, a_{33})$ on $\tilde{S}_1$, since $\tilde{S}_1$ is a finite set.

For the set of S_2, one of the elements a_{11}, a_{22}, a_{33} is at an endpoint while the other two elements are to be determined. Without loss of generality, we assume that a_{33} is an endpoint, i.e., $a_{33} = a_{33}^m$ *or* $a_{33} = a_{33}^M$. Then, $\frac{\partial f_i}{\partial a_{11}} = 0$ and $\frac{\partial f_i}{\partial a_{22}} = 0$, i.e., (4.7) and (4.8) hold. Since the left-hand sides of (4.7) and (4.8) are identical, so are the right-hand sides. This means that

$$a_{22}^2 = a_{11}^2 + a_{13}a_{31} - a_{23}a_{32}. \tag{4.13}$$

Combining (4.13) and (4.7), we obtain an equation for a_{11}. Letting x denote a_{11}, this equation assume the form

$$3x^4 + s_1x^3 + s_2x^2 + s_3x + s_4 = 0, \tag{4.14}$$

where s_1, s_2, s_3, and s_4 are the coefficients, which are calculated to be

$$\begin{aligned} s_1 &= 4a_{33} \\ s_2 &= -2a_{33}^2 + 2b_3 + 4b_2 - 2b_1 \\ s_3 &= -4a_{33}^3 + a_{33}(4b_3 + 8b_2 - 4b_1) \\ s_4 &= -a_{33}^4 + a_{33}^2(2b_3 + 4b_2 - 2b_1) - (b_3 + b_1)^2 \end{aligned}$$

with

$$b_1 = a_{23}a_{32}, \quad b_2 = a_{13}a_{31}, \quad b_3 = a_{12}a_{21}. \tag{4.15}$$

Expression (4.14) is a fourth order equation. Since any fourth order equation has a root formula, x can be determined precisely from equation (4.14), i.e., we are able to determine a_{11}, a_{22} precisely. This means that we can determine the finite set S_2 precisely.

We are therefore able to determine the extreme values of $f_i(a_{11}, a_{22}, a_{33})$ on the finite set G_i, given by

$$G_i = S_0 \cup \tilde{S}_1 \cup S_2, \tag{4.16}$$

where $S_0, \tilde{S}_1, S_2$ and are defined by (4.2), (4.12), (4.4) and , $1 \le i \le 64$. These extreme values are identical to the extreme values of $f_i(a_{11}, a_{22}, a_{33})$ on H.

Summarizing, we have proved the following result for the Hurwitz stability of 3 × 3 interval matrices.

Proposition 1 Assume that $[A^m, A^M]$ is a 3 × 3 real interval matrix and that A^M is Hurwitz stable. Then $[A^m, A^M]$ is Hurwitz stable if and only if the following conditions are satisfied:

(i) all corner matrices of $[A^m, A^M]$ have negative determinants and

(ii) $f_i(a_{11}, a_{22}, a_{33}) < 0$ for all $(a_{11}, a_{22}, a_{33}) \in G_i$, $1 \leq i \leq 64$. □

Let D_i denote the single point set in R^6 such that f_i is obtained from f by fixing the nondiagonal elements at D_i, and let

$$M = \cup_{i=1}^{64} G_i \times D_i, \tag{4.17}$$

where G_i is defined by (4.16). Then, M is a finite subset of $[A^m, A^M]$. Clearly, Proposition 1 and Corollary 1 imply the following result:

Theorem 2 A 3×3 real interval matrix $[A^m, A^M]$ is Hurwitz stable if and only if every matrix $A \in M$ is Hurwitz stable, where M is given by (4.17). □

It is straightforward to verify that M consists of at most $2^6(2^3 + 3 + 3 \cdot 4) = 1472$ matrices.

V. An Algorithm and Examples

Using Proposition 1, we now establish an algorithm to solve the Hurwitz-stability problem of 3×3 interval matrices.

Problem: Given an interval matrices $[A^m, A^M]$ in $R^{3\times 3}$, determine its Hurwitz stability properties.

Algorithm

1) Verify the Hurwitz stability of A^M. If A^M is not stable, stop.

2) Determine the determinants of all corner matrices in $[A^m, A^M]$. If any of them has non-negative determinant, stop.

3) Fix all nondiagonal elements a_{jk}, $1 \leq j \neq k \leq 3$ to their endpoints, i.e., a_{jk}^m or a_{jk}^M. We have a total of 64 functions, $f_i(a_{11}, a_{22}, a_{33})$, $1 \leq i \leq 64$ as described in section 5. Set $i = 1$.

4) Determine the values of $f_i(a_{11}, a_{22}, a_{33})$ for $(a_{11}, a_{22}, a_{33}) \in G_i$, where G_i is given by (4.16). If any of these values is nonnegative, stop.

5) If $i < 64$, let i be replaced by $i + 1$ and go to step 4. If $i = 64$, go to step 6.

6) The interval matrix is Hurwitz stable. □

Remarks

a) If the algorithm stops before it goes to step 6, the interval matrix is not Hurwitz-stable. Otherwise, it is Hurwitz stable.

b) For any 3×3 interval $[A^m, A^M]$, the algorithm terminates before or at step 6.

The above Algorithm has been implemented on a Sun SPARCstation 2 by using matlab. We demonstrate the the applicability of the Algorithm by the following examples. In each example, the process ends in 2 seconds (using time).

Example 1. The 3×3 interval matrix $[A^m, A^M]$, where

$$A^m = \begin{bmatrix} -3 & 4 & 4 \\ -4 & -6 & -4 \\ -5 & 2 & -5 \end{bmatrix} \text{ and } A^M = \begin{bmatrix} -2 & 5 & 6 \\ -3 & -2 & -3 \\ -4 & 3 & -2 \end{bmatrix}$$

has been determined to be Hurwitz stable by our algorithm.

Example 2. This example is given in Barmish [8] to provide a counter-example to the paper [9], where

$$A^m = \begin{bmatrix} -1.5 & -12.06 & -0.06 \\ -0.25 & 0 & 1 \\ 0.25 & -4 & -1 \end{bmatrix} \text{ and } A^M = \begin{bmatrix} -0.5 & -12.06 & -0.06 \\ -0.25 & 0 & 1 \\ 0.25 & -4 & -1 \end{bmatrix}.$$

This interval matrix is unstable since when $a_{11} = -1$, the function f defined in (4.1) has maximal value 0.06.

References

[1] J. D. Cobb, "The minimal dimension of stable faces required to guarantee stability of matrix polytope: D-stability", *Proc. 27th IEEE Conference on Decision and Control*, Austin, TX, pp.119-120, December, 1988.

[2] K. Wang and A. N. Michel, "On sufficient conditions for the stability of interval matrices", *Systems & Control Letters* , vol. 20, pp. 345-351, 1993.

[3] R. K. Miller and A. N. Michel, *Ordinary Differential Equations*, Academic Press, New York, 1982.

[4] T. Mori and H. Kokame, "Convergence property of interval matrices and interval polynomials", *International Journal of Control*, vol. 45, pp. 481-484, 1987.

[5] R. V. Patel and M. Toda, "Quantitative measures of robustness for multivariable systems", *1980 Joint Automatic Control Conference*, San Francisco, CA, TP8-A, August, 1980.

[6] R. K. Yedavalli, "Improved measures of stability robustness for linear state space models", *IEEE Transactions on Automatic Control*, vol. 30, pp. 577-579, 1985.

[7] J. R. Munkres *Topology, a First Course*, Englewood Cliffs, N.J., Prentice-Hall, 1974.

[8] B. Barmish and C.V. Hollot, "Counter-example to a recent result on the stability of interval matrices by S. Bialas", *International Journal of Control*, 1984, vol 39, pp. 1103-1105.

[9] S. Bialas, *International Journal of Control*, 1983.

[10] M. Mansour, "Robust stability of interval matrices", *Proc. 28th IEEE conference on Decision and Control*, Tampa, FL, pp. 46-51, 1989.

[11] K. Wang, A.N. Michel and D. Liu, "Necessary and sufficient conditions for the Hurwitz and Schur stability of interval matrices", *IEEE Transactions on Automatic Control*, July, 1994 (to appear).

[12] E. I. Jury, *Inners and Stability of Dynamical Systems*, Robert E. Krieger publishing company, 1982.

Exact Slow-Fast Decomposition of the $\mathcal{H}_\infty$ Control Riccati Equation of Singularly Perturbed Systems

A. H. Azzo and *M. E. Sawan*
The Wichita State University
Department of Electrical Engineering
Wichita, Ks 67260-0044

Abstract- *The $\mathcal{H}_\infty$ control Riccati equation, associated with state-feedback, of standard singularly perturbed systems is completely and exactly decomposed into two reduced-order $\mathcal{H}_\infty$ control Riccati equations corresponding to the slow and fast time scales. The pure-slow and pure-fast $\mathcal{H}_\infty$ control Riccati equations are nonsymmetric ones; however, their $O(\epsilon)$ perturbations are symmetric. It is shown that the Newton method is very efficient for solving the obtained nonsymmetric $\mathcal{H}_\infty$ control Riccati equations. The presented method is suited for parallel computations. Moreover, due to complete and exact decomposition of the Riccati equation , this procedure might produce new insight into the two-time scale $\mathcal{H}_\infty$ filtering and output feedback problems.*

1. Introduction and Problem Formulation

Let $\mathcal{P}$ denote a finite-dimensional and linear time-invariant plant described by

$$\mathcal{P} := \begin{cases} \dot{x} = A\,x + B_1\,w + B_2\,u, \\ z = C_1\,x + D_1\,u, \\ y = x\,, \end{cases} \tag{1}$$

where all the matrices in (1) are constant real matrices of compatible dimensions. Assume that:

the pair (C_1, A) has no unobservable modes on the imaginary axis, (2)

the pair (A, B_2) is stabilizable, (3)

$D_1^T[D_1 \quad C_1] = [0 \quad I].$ (4)

Let $\mathcal{A}(\mathcal{K})$ denote the set of finite-dimensional linear time-invariant internally stabilizing and proper controllers for $\mathcal{P}$. Given $K \in \mathcal{A}(\mathcal{K})$, we let $T_{zw}(K)$ denote the closed loop transfer matrix from w to z. The performance of the closed loop system is measured by

$$\gamma_\infty := \inf\{\, \|T_{zw}(K)\|_\infty : K \in \mathcal{A}(\mathcal{K}) \,\} \tag{5}$$

From [1] , the corresponding $\mathcal{H}_\infty$ control Riccati equation is given by

$$A^T X_\gamma + X_\gamma A + X_\gamma(1/\gamma^2 B_1 B_1^T - B_2 B_2^T)X_\gamma + C_1^T C_1 = 0 \tag{6}$$

In the special case when the system (1) is a standard singularly perturbed system, the matrices A, B_1, B_2, and C_1 take the special forms

$$A = \begin{bmatrix} A_{11} & A_{12} \\ \frac{1}{\epsilon}A_{21} & \frac{1}{\epsilon}A_{22} \end{bmatrix}, \qquad B_1 = \begin{bmatrix} B_{11} \\ \frac{1}{\epsilon}B_{12} \end{bmatrix}, \qquad B_2 = \begin{bmatrix} B_{21} \\ \frac{1}{\epsilon}B_{22} \end{bmatrix}, \qquad C_1 = [C_{11} \quad C_{12}] \tag{7}$$

Where ϵ is a small positive parameter and A_{22} is nonsingular at $\epsilon = 0$. As the parameter ϵ tends to zero, the solution behaves nonuniformly, producing a so-called stiff problem.

Let $Q = C_1^T C_1$, $S = (1/\gamma^2 B_1 B_1^T - B_2 B_2^T)$, L, T_1, T_2, T_3, and T_4 be decomposed as follows

$$Q = \begin{bmatrix} Q_1 & Q_2 \\ Q^T{}_2 & Q_3 \end{bmatrix}, \qquad L = \begin{bmatrix} L_1 & L_2 \\ L^T{}_2 & L_3 \end{bmatrix}, \qquad S = \begin{bmatrix} S_1 & \frac{Z}{\epsilon} \\ \frac{Z^T}{\epsilon} & \frac{S_2}{\epsilon^2} \end{bmatrix} \tag{8}$$

$$T_1 = \begin{bmatrix} A_1 & -S_1 \\ Q_1 & -A^T{}_1 \end{bmatrix}, \quad T_2 = \begin{bmatrix} A_2 & -Z \\ -Q_2 & -A_3{}^T \end{bmatrix}, \quad T_3 = \begin{bmatrix} A_3 & -Z^T \\ -Q_2{}^T & -A_2{}^T \end{bmatrix}, \quad T_4 = \begin{bmatrix} A_4 & -S_2 \\ -Q_3 & -A_4{}^T \end{bmatrix}$$

where L and H satisfy

$$T_4 L - T_3 - \epsilon L(T_1 - T_2 L) = 0 \tag{9}$$

$$-H(T_4 + \epsilon L_2) + T_2 + \epsilon(T_1 - T_2 L)H = 0. \tag{10}$$

The problem is to find the solution of the $\mathcal{H}_\infty$ control Riccati equation of Singularly perturbed systems (6) in terms of the solutions of reduced order pure-slow and pure-fast $\mathcal{H}_\infty$ control Riccati equations.

2. Main Results

The key point is to extend to the two-time-scale $\mathcal{H}_\infty$ control Riccati equation the decomposition technique applied in [2] to the two-time-scale algebraic Riccati equation, associated with the LQR problem.

Theorem 2.1 : *Let the plant $\mathcal{P}$ be a standard singularly perturbed system. Assume that* (2)-(4) *hold. Also, let A , Q, L and S be decomposed as in* (7), *and* (8). *Then , the following statements hold.*

1. *The full order $\mathcal{H}_\infty$ control Riccati equation* (6) *is decomposed into a nonsymmetric reduced-order slow algebraic Riccati equation corresponding to the slow time scale,*

$$\begin{aligned} &X_{\gamma 1}(A_1 - A_2 L_1 + Z L_3) + (A_1 - L^T{}_2 Q^T{}_2 - L^T{}_4 A_3)^T X_{\gamma 1} \\ &+ (Q_1 - Q_2 L_1 - A^T{}_3 L_3) - X_{\gamma 1}(S_1 - A_2 L_2 - Z L_4) X_{\gamma 1} = 0 \end{aligned} \tag{11}$$

and a nonsymmetric reduced-order fast algebraic Riccati equation corresponding to the fast time scale.

$$\begin{aligned} &X_{\gamma 2}(A_4 + \epsilon(L_1 A_2 + L_2 Q_2)) + (A^T{}_4 + \epsilon(L_3 Z + L_4 A^T{}_3)) X_{\gamma 2} \\ &+ (Q_3 - \epsilon(L_2 A_2 - L_4 Q_2)) - X_{\gamma 2}(S_2 + \epsilon(L_1 Z + L_2 A^T{}_3)) X_{\gamma 2} = 0 \end{aligned} \tag{12}$$

2. *The $O(\epsilon)$ approximations of (11) and (12) are symmetric*

$$A^T{}_{\gamma s} X_{\gamma s} + X_{\gamma s} A_{\gamma s} - X_s S_{\gamma s} X_{\gamma s} + Q_{\gamma s} = 0 \tag{13}$$

$$X_{\gamma 2} A_4 + A^T{}_4 X_{\gamma 2} + Q_3 - X_{\gamma 2} S X_{\gamma 2} = 0 \tag{14}$$

where $A_{\mu S}$, $Q_{\mu S}$, and $S_{\mu S}$ can be found in [3].

Proof: The proof is omitted for space limitation.

As mentioned in [2], Newton type-algorithm can be used efficiently for solving (11) and (12) having obtained good initial guesses from solving (13) and (14), respectively.

References

[1] J. Doyle, K. Glover, P. Khargonekar, and B. Francis, "State-space solutions to standard $\mathcal{H}_2$ and $\mathcal{H}_\infty$ control problems," IEEE Trans. Automat. Contr., vol. 34, no. 8, pp. 831-847, 1989.

[2] W. Su, Z. Gajic, and X. Shen, "Exact slow-fast decomposition of the algebraic Riccati equation of singularly perturbed systems," IEEE Trans. Automat. Contr., vol. 37, no. 9, pp. 1456-1459, 1992.

[3] J. H. Chow and P. V. Kokotovic, "A decomposition of near-optimum regulators for systems with slow and fast modes," IEEE Trans. Automat. Contr., vol. 21, no. 5, pp. 701-705, 1976.

On a Finite Horizon H_∞ Performance Robustness Problem

M. B. Subrahmanyam
Flight Control, Code 6053
Naval Air Warfare Center
Warminster, PA 18974-0591

SUMMARY

We present an approach for the solution of the finite horizon H_∞ performance robustness problem under parameter variations. For the detailed exposition of the results contained here, see Ref. 1. The n-dimensional time-varying system is given by

$$\dot{x} = A(t)x + B_1(t)u + B_2(t)v, \quad x(0) = 0, \tag{1}$$
$$z = C(t)x + D(t)u + E(t)v, \tag{2}$$
$$y = C_2(t)x + D_2(t)u + E_2(t)v. \tag{3}$$

The suboptimal performance robustness problem can be stated as follows. Given $\bar{\lambda}$, find an output feedback controller, if one exists, for which

$$\min_{v \neq 0} \frac{\int_0^T \frac{1}{2} v^* R v \, dt}{\int_0^T \frac{1}{2} z^* W z \, dt} > \bar{\lambda},$$

even under specified variations in the system matrices in (1)-(3). Here R and W are assumed to be positive definite and the superscript * denotes a matrix or vector transpose.

In Ref. 2, we gave equations for a suboptimal H_∞ design for a specified performance λ. Under suitable assumptions, a suboptimal H_∞ controller, if one exists, is given by

$$\dot{q} = Aq + B_1(U_1Pq + U_2q) + B_2(V_1Pq + V_2q) + L(\tilde{C}q + D_2u - y), \tag{4}$$
$$u = (U_1P + U_2)q. \tag{5}$$

For the details of derivation and the definition of various terms, see Ref. 2. In the case of time-invariant systems, the solutions P and Y of the associated Riccati equations eventually converge to constant matrices.

The closed loop system is given by

$$\begin{pmatrix} \dot{x} \\ \dot{q} \end{pmatrix} = \begin{pmatrix} A & B_1(U_1P + U_2) \\ -LC_2 & \tilde{U} - LD_2(U_1P + U_2) \end{pmatrix} \begin{pmatrix} x \\ q \end{pmatrix} + \begin{pmatrix} B_2 \\ -LE_2 \end{pmatrix} v, \tag{6}$$

where $\tilde{U}$ is as defined in Ref. 2. Note that all the matrices can be time-varying. We now specialize to the case where

$$x(0) = q(0) = 0. \tag{7}$$

Let

$$z = Cx + D(U_1P + U_2)q + Ev. \tag{8}$$

For the controller

$$u = (U_1P + U_2)q, \tag{9}$$

the performance is given by

$$\min_{v\neq 0} \frac{\int_0^T \frac{1}{2} v^* R v \, dt}{\int_0^T \frac{1}{2} z^* W z \, dt}. \tag{10}$$

The above value is strictly greater than λ by the design procedure given in Ref. 1. For time-invariant systems, it is convenient to consider the constant solutions of the two dynamic Riccati equations.

We now describe a procedure to compute the value of (10). Let (6) be rewritten as

$$\dot{x}_s = A_s x_s + B_s v, \quad x_s(0) = 0, \tag{11}$$

where v needs to be chosen to minimize

$$\frac{\int_0^T \frac{1}{2} v^* R v \, dt}{\int_0^T \{\frac{1}{2} x_s^* Q_1 x_s + x_s^* Q_2 v + \frac{1}{2} v^* Q_3 v\} \, dt}. \tag{12}$$

Note that the denominator of (10) can put in the form of the denominator of (12) by virtue of (8).

In the following theorem, we give conditions that are satisfied by an optimal v which minimizes (12) subject to (11).

THEOREM 1. *Consider the system given by (11) and (12). Let*

$$\hat{\lambda} = \min_{v\neq 0} \frac{\int_0^T \frac{1}{2} v^* R v \, dt}{\int_0^T \{\frac{1}{2} x_s^* Q_1 x_s + x_s^* Q_2 v + \frac{1}{2} v^* Q_3 v\} \, dt}. \tag{13}$$

Assume that $(R(t) - \hat{\lambda} Q_3(t))^{-1}$ exists for all $t \in [0, T]$. If (x_s, v) is optimal, then there exists a nonzero $\rho(t)$ such that

$$\frac{d\rho}{dt} = -A_s^* \rho - \hat{\lambda} Q_1 x_s - \hat{\lambda} Q_2 v, \quad \rho(T) = 0, \tag{14}$$

and

$$v(t) = (R - \hat{\lambda} Q_3)^{-1} \{\hat{\lambda} Q_2^* x_s + B_s^* \rho\}. \tag{15}$$

Let

$$\tilde{A} = A_s + \hat{\lambda} B_s (R - \hat{\lambda} Q_3)^{-1} Q_2^*, \tag{16}$$

$$\tilde{B} = B_s (R - \hat{\lambda} Q_3)^{-1} B_s^*, \tag{17}$$

and

$$\tilde{C} = -\hat{\lambda} Q_1 - \hat{\lambda}^2 Q_2 (R - \hat{\lambda} Q_3)^{-1} Q_2^*. \tag{18}$$

The variables satisfy a two point boundary value problem given by

$$\begin{pmatrix} \dot{x}_s \\ \dot{\rho} \end{pmatrix} = \begin{pmatrix} \tilde{A} & \tilde{B} \\ \tilde{C} & -\tilde{A}^* \end{pmatrix} \begin{pmatrix} x_s \\ \rho \end{pmatrix}, \tag{19}$$

with

$$x_s(0) = 0, \quad \rho(T) = 0. \tag{20}$$

We now give a criterion for the estimation of $\hat{\lambda}$. Notice that $\hat{\lambda}$ gives a measure of performance of the H_∞ optimal controller under worst-case conditions.

THEOREM 2. *Let $\hat{\lambda}$ be the smallest positive value for which the boundary value problem given by (19) and (20) has a solution (x_s, ρ) with $\int_0^T \{\frac{1}{2}x_s^* Q_1 x_s + x_s^* Q_2 v + \frac{1}{2}v^* Q_3 v\}\, dt > 0$, where $v \triangleq (R - \hat{\lambda}Q_3)^{-1}\{\hat{\lambda}Q_2^* x_s + B_s^* \rho\}$. Then $\hat{\lambda}$ is the minimum value of (12), x_s is an optimal trajectory and $v = (R - \hat{\lambda}Q_3)^{-1}\{\hat{\lambda}Q_2^* x_s + B_s^* \rho\}$ is an open loop optimal exogenous input.*

Note that the boundary value problem (19)-(20) has a solution with a nonvanishing denominator for (12) for at most a countably infinite values of $\hat{\lambda}$. Theorem 2 gives a sufficient condition for an open loop exogenous input to be optimal. Theorems 1 and 2 completely characterize the open loop worst case exogenous input.

We now develop a formula for the variation of $\hat{\lambda}$ when there are parameter variations in the system matrices. The system is given by

$$\dot{x}_s = A_s x_s + B_s v, \quad x_s(0) = 0, \tag{21}$$

with v chosen to minimize

$$\frac{\int_0^T \frac{1}{2} v^* R v \, dt}{\int_0^T \{\frac{1}{2}x_s^* Q_1 x_s + x_s^* Q_2 v + \frac{1}{2}v^* Q_3 v\}}. \tag{22}$$

From (19)-(20), we have a boundary value problem given by

$$\dot{x}_s = \tilde{A} x_s + \tilde{B} \rho, \tag{23}$$

$$\dot{\rho} = \tilde{C} x_s - \tilde{A}^* \rho, \tag{24}$$

with

$$x_s(0) = 0, \quad \rho(T) = 0. \tag{25}$$

Note that $\hat{\lambda}$ is the minimal value of (22). Because of variations in the matrices of the original system (1)-(3), there will be corresponding variations in the matrices $\tilde{A}(t), \tilde{B}(t)$, and $\tilde{C}(t)$. Let the elemental dependent variations in $\tilde{A}$, $\tilde{B}$, and $\tilde{C}$ be denoted by $\delta\tilde{A}$, $\delta\tilde{B}$, and $\delta\tilde{C}$ respectively. For simplicity of notation, we will derive the variation in performance in terms of $\delta\tilde{A}$, $\delta\tilde{B}$, and $\delta\tilde{C}$. We denote by $\delta\hat{\lambda}$ the variation in $\hat{\lambda}$ owing to the parameter variations.

Let

$$\Upsilon = (R - \hat{\lambda}Q_3)^{-1}. \tag{26}$$

We have

$$\delta\tilde{A} = I_1 + \delta\hat{\lambda} J_1, \tag{27}$$

$$\delta\tilde{B} = I_2 + \delta\hat{\lambda} J_2, \tag{28}$$

$$\delta\tilde{C} = I_3 + \delta\hat{\lambda} J_3, \tag{29}$$

where

$$\begin{aligned}
I_1 &= \delta A_s + \hat{\lambda}\delta B_s \Upsilon Q_2^* + \hat{\lambda} B_s \Upsilon(\delta R - \hat{\lambda}\delta Q_3)\Upsilon Q_2^* + \hat{\lambda} B_s \Upsilon \delta Q_2^*,\\
J_1 &= -\hat{\lambda} B_s \Upsilon Q_3 \Upsilon Q_2^* + B_s \Upsilon Q_2^*,\\
I_2 &= \delta B_s \Upsilon B_s^* + B_s \Upsilon(\delta R - \hat{\lambda}\delta Q_3)\Upsilon B_s^* + B_s \Upsilon \delta B_s^*,\\
J_2 &= -B_s \Upsilon Q_3 \Upsilon B_s^*,\\
I_3 &= -\hat{\lambda}\delta Q_1 - \hat{\lambda}^2 \delta Q_2 \Upsilon Q_2^* - \hat{\lambda}^2 Q_2 \Upsilon(\delta R - \hat{\lambda}\delta Q_3)\Upsilon Q_2^* - \hat{\lambda}^2 Q_2 \Upsilon \delta Q_2^*,\\
J_3 &= -Q_1 - 2\hat{\lambda} Q_2 \Upsilon Q_2^* + \hat{\lambda}^2 Q_2 \Upsilon Q_3 \Upsilon Q_2^*.
\end{aligned}$$

THEOREM 3. *The variation $\delta\hat{\lambda}$ in performance is given by*

$$\delta\hat{\lambda} = \frac{-\int_0^T \{x_s^* I_1^* \rho + \frac{1}{2}\rho^* I_2 \rho - \frac{1}{2}x_s^* I_3 x_s\}\, dt}{\int_0^T \{\frac{1}{2}x_s^* Q_1 x_s + x_s^* Q_2 v + \frac{1}{2}v^* Q_3 v\}\, dt}, \tag{30}$$

where $v = \Upsilon\{B_s^* \rho + \hat{\lambda} Q_2^* x_s\}$.

We now give an iterative procedure to solve the performance robustness problem. The suboptimal performance robustness problem is to find a controller, if one exists, for which

$$\min_{v \neq 0} \frac{\int_0^T \frac{1}{2} v^* R v\, dt}{\int_0^T \frac{1}{2} z^* W z\, dt} > \bar{\lambda}, \tag{31}$$

under variations in the system matrices involved in (1)-(3).

The iterative steps are as follows.

1. Set $\lambda = \lambda_i$ and design a suboptimal H_∞ controller based on the design equations of Ref. 2.
2. For this controller, let

$$\hat{\lambda} = \min_{v \neq 0} \frac{\int_0^T \frac{1}{2} v^* R v\, dt}{\int_0^T \{\frac{1}{2}x_s^* Q_1 x_s + x_s^* Q_2 v + \frac{1}{2} v^* Q_3 v\}\, dt}. \tag{32}$$

   This can be computed using the results of Theorem 2. Note that $\hat{\lambda} > \lambda$ in many problems even when the time dependence of the matrices P and Y is ignored. If $\hat{\lambda} \leq \bar{\lambda}$, increase the value of λ and go through Steps 1 and 2 until $\hat{\lambda} > \bar{\lambda}$.
3. For the range of allowable variations of the matrices involved in (1)-(3), find the worst value of $\delta\hat{\lambda}$ using (30). Note that (30) is linear in the parameter variations. Thus, let $\delta = \min \delta\hat{\lambda}$. The controller of Step 1 solves the performance robustness problem in case $\hat{\lambda} + \delta > \bar{\lambda}$. If this is the case, stop the iteration and use the controller of Step 1.
4. If $\hat{\lambda} + \delta \leq \bar{\lambda}$, choose a suitable λ_{i+1} and set $\lambda = \lambda_{i+1}$. Go back to Step 1. Note that λ_{i+1} may be bigger or smaller than λ_i depending on the specific problem being solved.

Alternately we can get $\hat{\lambda}$ and $\delta = \min \delta\hat{\lambda}$ for a range of values of λ and then pick a value of λ that solves the performance robustness problem. In case we cannot find a value of λ which satisfies the performance robustness requirement, then the requirement is too stringent and the performance robustness problem does not have a solution. The requirement can be relaxed either by lowering $\bar{\lambda}$ or by suitably shrinking the range of allowable parameter variations.

REFERENCES

1. SUBRAHMANYAM, M. B., 'Finite Horizon H_∞ with Parameter Variations', *International Journal of Robust and Nonlinear Control*, To appear.
2. SUBRAHMANYAM, M. B., 'General formulae for suboptimal H_∞ control over a finite horizon', *International Journal of Control*, **57**, 365-375 (1993).

$\mathcal{L}_2$ to $\mathcal{L}_\infty$ Gains for Sampled-Data Systems

Guoming Zhu and Robert E. Skelton
Space Systems Control Laboratory, Potter 374
Purdue University, West Lafayette, IN 47907

Introduction

The $\mathcal{L}_2$ to $\mathcal{L}_\infty$ gains of continuous and discrete time-invariant systems are provided in [2,3]. Reference [1] provides the $\mathcal{L}_2$ to $\mathcal{L}_\infty$ gains of sampled-data systems, formulated by lifting operators. However, the numerical computation of those gains is not attractive (an infinite summation of lift operators must be computed). This paper allows the $\mathcal{L}_2$ to $\mathcal{L}_\infty$ gains of sampled-data systems to be computed directly by analytical formulae. The sampled-data system in this paper consists of a continuous plant $\mathcal{P}$, a discrete controller $\mathcal{C}$, a delay operator $\mathcal{D}_\delta$ such that $\mathcal{D}_\delta u(t) = u(t-\delta)$, where $0 \leq \delta < h$, a sampler $\mathcal{S}$ with period h, and a hold $\mathcal{H}$. The continuous plant has the following state space form

$$\mathcal{P}: \ \dot{x}_p = A_p x_p + B_p \mathcal{D}_\delta u + D_p w_p \ ; \ y = C_p x_p \ ; \ z = M_p x_p + L v \ , \tag{1}$$

where $x \in \mathcal{R}^{n_x}$, $u \in \mathcal{R}^{n_u}$, $w_p \in \mathcal{R}^{n_w}$, $y \in \mathcal{R}^{n_y}$, $z \in \mathcal{R}^{n_z}$, and $v \in \mathcal{R}^{n_z}$ are system state, control, exogenous input, performance output, system measurement and measurement noise vectors, respectively. The discrete-time controller $\mathcal{C}$ is in the following form

$$\mathcal{C}: \ x_c[(k+1)h] = A_c x_c(kh) + B_c z(kh) \ ; \ u(kh) = C_c x_c(kh) + D_c z(kh) \ , \tag{2}$$

where h is the sample period of the discrete-time controller $\mathcal{C}$.

Now define an operator m such that $\forall \ t \in \mathcal{R}$, $m[t] = (k, \sigma)$, where $k = \lfloor t/h \rfloor$, $\sigma = t - \lfloor t/h \rfloor h$, and $\lfloor e \rfloor$ equals to the integer part of e. Hence, m is a mapping from $\mathcal{R}$ to $\mathcal{I} \times [0, h)$, where $\mathcal{I}$ is the set of integers. Throughout this paper, we use $y(k,\sigma)$ to denote $y(m^{-1}[(k,\sigma)])$ for convenience, where $m^{-1}[k,\sigma] = kh + \sigma = t$. Therefore, $y(k,\sigma) = y(kh+\sigma)$. Note that $y(k,\sigma)$ is a discrete signal. Four kinds of $\mathcal{L}_\infty$ norms of output y are defined below

$$\begin{aligned} &||y(\cdot,\sigma)||_{\infty,2} \triangleq \sup_{k\geq 0}[y^T(k,\sigma)y(k,\sigma)]^{1/2} \ ; \ ||y||_{\infty,2} \triangleq \sup_{t\geq 0}[y^T(t)y(t)]^{1/2}; \\ &||y(\cdot,\sigma)||_{\infty,\infty} \triangleq \sup_{k\geq 0} \max_{1\leq j\leq n_y} |y_j(k,\sigma)| \ ; \ ||y||_{\infty,\infty} \triangleq \sup_{t\geq 0} \max_{1\leq j\leq n_y} |y_j(t)|. \end{aligned} \tag{3}$$

where y_j is the j-th entry of y. It is obvious that $||y||_{\infty,2} = \sup_{0\leq\sigma<h} ||y(\cdot,\sigma)||_{\infty,2}$.

The input w_p is assumed to be $\mathcal{L}_2$ signals, i.e., $w_p \in \mathcal{L}_2[0,\infty)$. Note that measurement disturbance v is sampled and held through controller $\mathcal{C}$. Only its sampled (and held) values appear to the system, due to the action of digital sensors. We denote this sample and hold operation by signal, $\bar{v} = \mathcal{H}\mathcal{S}v$, where $\mathcal{H}$ is a zero order hold operator. Assume that signal $\mathcal{S}v$ belongs to l_2, then, $\bar{v} \in \mathcal{L}_2[0,\infty)$. Define a subspace of $\mathcal{L}_2$

$$\bar{\mathcal{L}}_2 \triangleq \{[w_p^T, \bar{v}^T]^T \ : \ w_p \in \mathcal{L}_2 \ and \ \bar{v} = \mathcal{H}\mathcal{S}v \ ; \ \mathcal{S}v \in l_2\} \ . \tag{4}$$

The input of the sampled-data system is $w = [w_p^T, \bar{v}^T]^T$, whose $\mathcal{L}_2$ norm is defined by

$$||w||_2 \triangleq \{\int_0^\infty w^T w \ dt\}^{1/2} = \{\int_0^\infty w_p^T(t) w_p(t) dt + h \sum_{k=0}^{\infty} v^T(kh) v(kh) \ \}^{1/2}, \tag{5}$$

Here, the $\mathcal{L}_2$ to $\mathcal{L}_\infty$ gains of the sampled-data system refer to the $\bar{\mathcal{L}}_2$ inputs. In fact the $\mathcal{L}_2$ to $\mathcal{L}_\infty$ gains would be unbounded above if the inputs are referred to $\mathcal{L}_2$ signals. Four kinds of the $\mathcal{L}_2$ to $\mathcal{L}_\infty$ gains are defined below

$$\begin{aligned} &G_2^{\infty,2}(\sigma) \triangleq \sup_w\{||y(\cdot,\sigma)||_{\infty,2} : ||w||_2 = 1\} \ ; \ \bar{G}_2^{\infty,2} \triangleq \sup_w\{||y||_{\infty,2} : \ ||w||_2 = 1\}; \\ &G_2^{\infty,\infty}(\sigma) \triangleq \sup_w\{||y(\cdot,\sigma)||_{\infty,\infty} \ : \ ||w||_2 = 1\} \ ; \ \bar{G}_2^{\infty,\infty} \triangleq \sup_w\{||y||_{\infty,\infty} \ : \ ||w||_2 = 1\}. \end{aligned} \tag{6}$$

$\mathcal{L}_2$ to $\mathcal{L}_\infty$ Gains

All four kinds of the $\mathcal{L}_2$ to $\mathcal{L}_\infty$ Gains can be expressed by analytical formulas which are provided in this section.

Theorem 1 *For the asymptotically stable sampled-data system, define the l_∞ norm of output $y(k,\sigma)$ by (3) and the $\mathcal{L}_2$ norm of input w by (5). Let X be the unique positive semidefinite solution to the following discrete Lyapunov equation*

$$X = AXA^T + \overline{W}\,, \tag{7}$$

Then the $\mathcal{L}_2$ to l_∞ gain $G_2^{\infty,2}(\sigma)$ of the sampled-data system is

$$G_2^{\infty,2}(\sigma) = \bar{\sigma}^{1/2}[Y(\sigma)]\ ;\ Y(\sigma) = CA_\sigma X A_\sigma^T C^T + CW_\sigma C^T\,, \tag{8}$$

where matrices A, $\bar{W}$, C, A_σ and W_σ are defined below

$$A_\sigma = \begin{bmatrix} A_p^\sigma + B(\delta,\sigma)D_cM_p & B_\sigma^\delta & B(\delta,\sigma)C_c \\ 0 & 0 & 0 \\ 0 & 0 & I \end{bmatrix};\ C^T = \begin{bmatrix} C_p^T \\ 0 \\ 0 \end{bmatrix} \tag{9a}$$

$$A = \begin{bmatrix} A_p^h + B(\delta,h)D_cM_p & B(0,\delta) & B(\delta,h)C_c \\ D_cM_p & 0 & C_c \\ B_cM_p & 0 & A_c \end{bmatrix}; \tag{9b}$$

$$\overline{W} = DD^T/h + \begin{bmatrix} \int_0^h e^{A_p(h-\tau)}D_pD_p^Te^{A_p^T(h-\tau)}d\tau & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix};\ D = \begin{bmatrix} B(\delta,h)D_c \\ D_c \\ B_c \end{bmatrix} L\,; \tag{9c}$$

$$W_\sigma = D_\sigma D_\sigma^T/\sigma + \begin{bmatrix} \int_0^\sigma e^{A_p(\sigma-\tau)}D_pD_p^Te^{A_p^T(\sigma-\tau)}d\tau & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix};\ D_\sigma = \begin{bmatrix} B(\delta,\sigma)L \\ 0 \\ 0 \end{bmatrix}, \tag{9d}$$

and $A_\sigma = e^{A_p\sigma}$, $A_p^h = e^{A_ph}$, $B(\delta,\sigma) = \int_\delta^\sigma e^{A_p(\sigma-\tau)}B_pd\tau$ (if $\sigma \le \delta, B(\delta,\sigma) = 0$), $B_\sigma^\delta = \int_0^\delta e^{A_p(\sigma-\tau)}B_pd\tau$ if $\sigma > \delta$ ($\int_0^\sigma e^{A_p(\sigma-\tau)}B_pd\tau$ if $\sigma \le \delta$).

Theorem 2 *For the asymptotically stable sampled-data system, define the $\mathcal{L}_\infty$ norm of output y to be (3). Then the $\mathcal{L}_2$ to $\mathcal{L}_\infty$ gain of the sampled-data system is $\bar{G}_2^{\infty,2} = \sup_{0\le\sigma<h} G_2^{\infty,2}(\sigma)$, where the $\mathcal{L}_2$ norm of w is defined in (5).*

Remark 1 *The $\mathcal{L}_2$ to $\mathcal{L}_\infty$ gains $G_2^{\infty,\infty}(\sigma)$ and $\bar{G}_2^{\infty,\infty}$ are*

$$G_2^{\infty,\infty}(\sigma) = \max_{1\le j\le n_y}[Y(\sigma)]_{jj}\ ;\ \bar{G}_2^{\infty,\infty} = \max_{1\le j\le n_y}\sup_{0\le\sigma<h}[Y(\sigma)]_{jj}\,, \tag{10}$$

where $[\cdot]_{ii}$ represents the i-th diagonal entry of $[\cdot]$, and $Y(\sigma)$ is defined in Theorem 1.

Conclusions

Four kinds of $\mathcal{L}_2$ to $\mathcal{L}_\infty$ gains for sampled-data systems are defined and derived in this note. It turns out that these gains can be expressed by analytical formulae which are easy to compute (only a discrete Lyapunov equation is involved).

Reference

1. B. Bamieh, J. B. Pearson, B. A. Francis, and A. Tannenbaum. A lifting technique for linear periodic systems with applications to sampled-data control. *Syst. & Contr. Letters*, 17(1), 1991.
2. D. A. Wilson. Convolution and Hankel operator norms for linear systems. *IEEE Trans. Automat. Contr.*, 34(1), 1989.
3. G. Zhu, M. Corless, and R. Skelton. Robustness properties of covariance controllers. In *Proceeding of Allerton Conf.*, Monticello, IL., September 1989.

SIMULTANEOUS AND INTERVAL-PLANT STABILIZATION IN DISCRETE-TIME

Engin Yaz
Electrical Engineering Department, University of Arkansas, Fayetteville, AR, 72701

Summary

This work is on static output feedback stabilization of both a finite and an infinite family of discrete-time plants. The proposed technique relies on a combination of covariance assignment theory with a result on the common solution of linear matrix equations which allows one to tackle the problem with a finite step iterative method for both checking the existence of and generating the stabilizing controller.

Main Results

Consider the family of n_p discrete-time plants given below

$$x_{k+1} = A^i x_k + B^i u_k + {v_k}^i, \quad y_k = C^i x_k + {w_k}^i \tag{1}$$

where $i = 1, \ldots, n_p$, $x_k \in \mathbf{R}^{n_x}$ are the states, $u_k \in \mathbf{R}^{n_u}$ the controls, and $y_k \in \mathbf{R}^{n_y}$ are the outputs of these plants. Although the dimensions of all the members of this family are assumed to be the same, each member has a different set of coefficient matrices A^i, B^i, C^i and noise statistics $V^i > 0$ and $W^i > 0$ where V^i and W^i are respectively the covariances of white, zero mean, mutually uncorrelated (and independent of x_o) noises ${v_k}^i$ and ${w_k}^i$.

The problem we will first tackle is finding a single static feedback gain K such that the output control $u_k = K y_k$ simultaneously asymptotically stabilizes the zero solution of all the plants in this family in unforced form. We will employ a covariance assignment type control method to achieve this aim. Writing the closed loop state covariances (steady state second moments), we obtain

$$X^i = (A^i + B^i K C^i) X^i (A^i + B^i K C^i)^T + B^i K W^i K^T {B^i}^T + V^i \tag{2}$$

The stochastic stationary property which is equivalent to having constant covariances is valid if and only if all $(A^i + B^i K C^i)$ are Schur matrices. So, constant covariance assignment to all these plants by a single gain K will guarantee simultaneous stability.

Define

$$\Phi(X^i) := X^i - A^i X^i (A^i)^T - V^i + A^i X^i (C^i)^T (C^i X^i (C^i)^T + W^i)^{-1} C^i X^i (A^i)^T := L^i (L^i)^T \tag{3}$$

where Φ is assumed to be nonnegative definite,

$$\begin{aligned} T^i (T^i)^T &:= C^i X^i (C^i)^T + W^i, \ (K^o)^i := A^i X^i (C^i)^T (T^i (T^i)^T)^{-1} \\ D^i &:= L^i U^i (T^i)^{-1} - (K^o)^i \end{aligned} \tag{4}$$

for orthogonal U^i.

Theorem 1. State covariances $X^i > 0$ are assignable to matrix equations (2) by a single gain K if and only if $\Phi(X^i) = L^i (L^i)^T$ for $L^i \in \mathbf{R}^{n_x \times n_y}$ and

$$\begin{aligned} 0 = {} & [I - E^i (E^i)^\dagger][X^i - A^i X^i {A^i}^T - V^i + B^i G^{i-1} C^i X^i {A^i}^T + A^i X^i {C^i}^T (G^{i-1})^T {B^i}^T \\ & + B^i G^{i-1} (C^i X^i {C^i}^T + W^i)(G^{i-1})^T {B^i}^T][I - E^i {E^i}^\dagger] \end{aligned} \tag{5}$$

hold for $i = 1, ..., n_p$. If $X^i > 0$ are assignable, then the assigning K are given by

$$K = G^{n_p} + H^{n_p} Z \tag{6}$$

where Z is arbitrary and

$$\begin{aligned} &E^1 = B^1,\ F^1 = D^1,\ G^0 = 0, H^0 = I \\ &E^i = B^i H^{i-1},\ F^i = D^i - B^i G^{i-1},\ G^i = G^{i-1} + H^{i-1}(E^i)^\dagger F^i, \\ &H^i = H^{i-1}(I - [E^i]^\dagger E^i) \end{aligned} \tag{7}$$

with the orthogonal matrices being found by

$$\begin{aligned} U^i &= \mathcal{V}_1{}^i \mathrm{blockdiag}[\mathrm{I}_{\mathrm{p_i}}, \bar{\mathrm{U}}^{\mathrm{i}}] \mathcal{V}_2{}^{\mathrm{iT}} \\ \mathcal{W}_1{}^i \Sigma^i \mathcal{V}_1{}^i &= [I - E^i(E^i)^\dagger] L^i \\ \mathcal{W}_1{}^i \Sigma^i \mathcal{V}_2{}^{iT} &= [I - E^i(E^i)^\dagger](K^{oi} + B^i G^{i-1}) T^i \end{aligned} \tag{8}$$

where p_i is the rank of the corresponding matrix and $\overline{U}^i$ is arbitrary orthogonal. The above theory can be applied to stabilization of interval-plants

$$x_{k+1} = A_I x_k + B_I u_k + v_k,\ \ y_k = C_I x_k + w_k \tag{9}$$

where

$$A_I = \{A(\pi) :\ A(\pi) = \sum_{i=1}^{n_p} \pi_i A^i,\ \ \pi_i \geq 0,\ \sum_{i=1}^{n_p} \pi_i = 1\} \tag{10}$$

with B_I and C_I defined likewise, by using an interval-plant stability result in [1] based on the simultaneous stabilization of vertex parameters by a common quadratic Lyapunov function similar to the approach taken in [2] for continuous time models.

ACKNOWLEDGEMENTS

The author gratefully acknowledges stimulating discussions on this subject with Professor Robert Skelton and Mr. Karolos Grigoriadis from Purdue University.

REFERENCES

1. T.Mori and K.Kokame,"Convergence property of interval matrices and interval polynomials," Int. J. Control, vol.45, pp.481-484, 1987.
2. K.Grigoriadis, E.Yaz, and R.E.Skelton,"Robust stabilization of interval plants by state space methods,"Proc. of Allerton Conf. Communic., Contr., and Computing, Monticello, IL, pp.359-368, 1992.

Optimal Multiplexing on a Single Link: Delay and Buffer Requirements

Leonidas Georgiadis, Roch Guerin and Abhay Parekh[1]

Abstract. We study an important issue in the context of integrated network, the relationship between buffer and delay requirements for different traffic classes. Our focus is on the ability of *scheduling policies* to meet delay and buffer requirements individually and jointly in the context of a single link. We employ a commonly used characterization of traffic classes to establish the optimality of various policies. This covers aspects of schedulability and buffer sizing under different memory allocation policies. In addition, we also address the problem of fairness in the case of general packet arrival patterns with soft deadlines. The results of this paper provide a framework within which to address the design of service policies in integrated networks that must support traffic classes with different performance requirements.

1 Introduction

Central to the effective operation of an integrated network is its ability to estimate and control packet delay and buffer requirements for a wide range of traffic classes. These two parameters are related, and it is essential to understand the fundamental nature of this relationship in order to provide desired levels of performance without under-utilizing the network. The lack of a suitable framework within which to analyze delay and buffer requirements, and the pressing need for effective mechanisms for sharing buffers and links in the network, have led to a number of service disciplines and buffer management policies [10, 14, 17] that are difficult to compare. In this paper we focus on understanding optimal policies for delay and buffer management in the context of a single link.

Assume a fluid flow of traffic into a multiplexer from K input links, where the speed of the i^{th} link is r_i, and partition the flow from each input link into discrete entities or packets. A packet may be infinitesimally small, but can be no larger than $L_{\max}$ bits. Arriving bits are stored in the memory of the multiplexer until they are transmitted on the output link, which is assumed to be of speed r. A packet becomes eligible for transmission only after its last bit has arrived at the multiplexer. A scheduling policy decides which of the eligible packets to transmit on the output link and then transmits this packet non-preemptively. We will only study *zero-loss* multiplexers, i.e. the memory is appropriately sized so that there is always space to store a bit arriving on any input link.

In integrated data networks that must support a wide range of performance levels it is important for the multiplexer to discriminate among different traffic classes. Let there be N classes and let $A_{ji}(\tau, t)$ be the number of class i bits that arrive on link j in the interval $[\tau, t)$. Also, let

$$A_i(\tau,t) = \sum_{j=1}^{K} A_{ji}(\tau,t)$$

be the number of class i bits that arrive in the interval $[\tau, t)$. For each i, set $A_i(\tau, t+\tau) = 0$ for $\tau < 0$. Let there exist $\sigma_i, \rho_i \geq 0$ such that

$$A_i(\tau,t) \leq \sigma_i + \rho_i(t-\tau) \tag{1}$$

and

$$\sum_{i=1}^{N} \rho_i \leq r. \tag{2}$$

---

[1] T. J. Watson Research Center, IBM

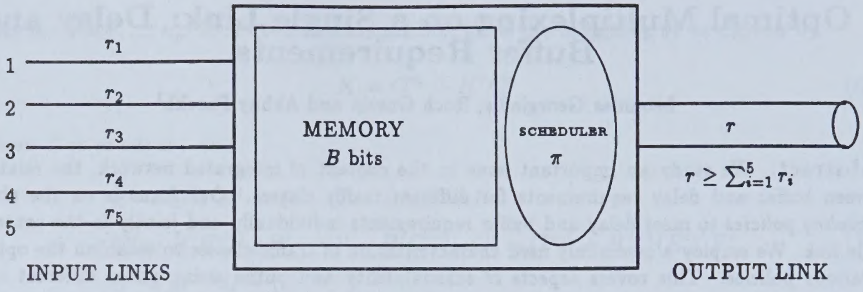


Figure 1: The Multiplexer Model when $\sigma_i = 0$ for $i = 1, 2, ..., N$.

This model for incoming traffic is identical to the one proposed by Cruz [5], [6]. Note that the arrival constraints can be used to model the input links of a multiplexer. Let the speed of the i^{th} input link be r_i. Then setting, $N = K$, $\sigma_i = 0$, $\rho_i = r_i$ for $i = 1, 2, , .K$, each class is associated with a distinct input link, and the conditions (1) and (2) reduce to the natural condition $\sum_{j=1}^{K} r_j \leq r$.

For sample path arguments we will denote $S_i^\pi(0, t)$ to be the number of class i bits served in the interval $[0, t)$ and $Q_i^\pi(t)$ to be the number of class i bits stored at time t, when the scheduler implements policy π. For a zero-loss multiplexer that implements the service policy π, define for each class i:

$$M_i^\pi = \sup_{t \geq 0,\ \text{arrivals}} Q_i^\pi(t), \tag{3}$$

where the arrivals are consistent with (1) and (2).

Define the delay of a packet to be the time it spends in the system, i.e. the sum of the time that it spends waiting in the memory and the time taken to transmit it on the output link. Given a zero-loss multiplexer that implements service policy π, let $D = (D_1, ..., D_N)$ be a *schedulable* vector if for all arrival patterns consistent with (1) and (2), and for all classes i, no class i packet is delayed by more than D_i time units. The *schedulable region* Ω^π of the policy π is the set of all vectors schedulable under π. Define a scheduling policy π^* to be delay-optimal among a class of admissible policies $\mathcal{A}$ if when applied to a zero-loss multiplexer:

$$\Omega^\pi \subset \Omega^{\pi^*} \tag{4}$$

for all policies $\pi \in \mathcal{A}$. Also, for policy π and class i, let D_i^π be the smallest real number such that no class i packet is delayed by more than D_i^π over all feasible arrival patterns. Thus D_i^π is the best possible upper bound on the class i delay under policy π over all arrival patterns.

So far we have assumed that the size of the multiplexer memory, B^π, is large enough to allow for zero loss. However, as suggested by (3), the sizing of the buffer space is dependent on the service policy and therefore also on the delays. Another important factor that affects the sizing of the buffers is the flexibility of the buffer allocation mechanism (the function of assigning memory locations to arriving packets). We consider three natural ways in which the buffers can be structured:

1. Flexible Allocation: Packets from all arrival streams share a common pool of memory, i.e. buffers are not allocated by class. This is difficult to implement since it requires multiple parallel writes to be implemented by a single control logic, and also a linked list structure to maintain packet order. Note that $B^\pi \geq \sup_{t \geq 0,\ \text{arrivals}} \sum_{i=1}^{N} Q_i^\pi(t)$.

2. Semi-Flexible Allocation: There are B_i^π bits of buffer allocated to packets from class i. The value of B_i^π cannot be changed after $t = 0$. This requires the multiplexer to be programmable so that the allocations can match the traffic class characteristics, however, the link list structure becomes simpler to implement than in the case of Semi-Flexible allocation. Also, the multiple

parallel writes can be implemented by multiple control logic modules. Note that $B_i^\pi \geq M_i^\pi$ and that therefore $B^\pi \geq \sum_{i=1}^{N} M_i^\pi$.

3. Fixed Allocation: There are b bits of buffer allocated to packets from each class i, i.e., $M_i^\pi = b$ for all i. Here $B^\pi \geq N \max_{i=1,2,\dots,N} M_i^\pi$.

In some cases, it may be desirable to do the buffer allocation by *link* as opposed to class. Recall that this can be incorporated into the model by defining $N = K$ and setting $\rho_i = r_i$, $\sigma_i = 0$ for $i = 1, 2, \dots, N$.

Given $\alpha \in$ {Flexible, Semi-Flexible, Fixed}, policy π^* is buffer-optimal policy among the class of admissible policies $\mathcal{A}$, if when applied to a multiplexer with memory size $B_{\min}(\alpha)$ results in no loss and $B_{\min}(\alpha)$ is such that

$$B^\pi \geq B_{\min}(\alpha) \tag{5}$$

for any other policy $\pi \in \mathcal{A}$ that follows method α.

1.1 Outline

In Section 2 we introduce the notion of non-preemptive Tracking Policies, which we use throughout the paper. These policies are derived in a systematic way from any preemptive policy, but the order in which the packets are scheduled can differ *significantly* from other (oblivious) non-preemptive versions of these policies. However, when applied to Preemptive Earliest Deadline First policies (PEDF), and under *fixed* packet sizes, we show that the tracking (T(PEDF)) and oblivious non-preemptive policies (NPEDF) behave *identically.* In Section 3 we show that T(PEDF) and NPEDF are both delay-optimal among the class of non-preemptive policies, even though (under variable packet sizes) the order in which they schedule packets can be markedly different. We then obtain the optimal schedulable region, and compare it to that of the PEDF policy, which is delay optimal over the class of preemptive policies.

In Sections 4 and 5 we consider two figures of merit that attempt to minimize the lateness of packets in a "fair" manner. The first is that of minimizing the maximum lateness over all packets under any arrival pattern. The optimality of the preemptive EDF policy for this criterion has been established earlier ([8]), and it can be easily shown that NPEDF is non-optimal. However, we establish that under NPEDF and T(PEDF) maximum lateness is no more than $L_{\max}/r$ time units greater than it is under PEDF, and that thus these policies are very close to being optimal. The second, and stronger figure of merit is the degree to which the packet lateness vector is close to being lexicographically minimal. We show that a particular version of PEDF, which we call PEDF* is lexicographically optimal among all preemptive policies. Further, we show that the tracking policy, T(PEDF*), is close to being lexicographically optimal in that no packet is delayed by more than $L_{\max}/r$ under T(PEDF*) than it is under PEDF*.

In Section 6, we consider buffer-optimal policies. Under flexible allocation it follows easily that all work-conserving policies are buffer optimal. Adaptation of earlier work in [15] yields the buffer optimal policy in the fixed allocation case. Our main contribution in this section is the analysis of semi-flexible case, where we find that optimal buffer requirements are about the same as that of fixed allocation. This is surprising and seems to suggest that it is always preferable to implement fixed over semi-flexible allocation. However, the *schedulable* regions of buffer optimal policies are much larger under semi-flexible allocation than under fixed allocation. In fact, we have recently shown that one can design delay-optimal policies that are buffer-optimal under semi-flexible allocation. In Section 7 we demonstrate in that the design of such policies is not possible under fixed allocation.

Basic relationships between delay and buffer optimality are illustrated in Section 7, and our conclusions are in Section 8.

1.2 Prior Work

The problem of scheduling tasks has received significant attention in the context of (real-time) computing systems, where important results on optimal scheduling policies and of the associated schedulable region have been obtained. However many of the results assume more restrictive arrival functions than those given by (1) and (2): The optimality of the PEDF for the class of preemptive policies was first shown in [13] for periodic arrivals. In [11, 12] the delay-optimality of NPEDF among the class of non-preemptive policies is established for periodic and so-called sporadic arrivals. The schedulable regions for NPEDF and PEDF have been derived in [18] for arrival streams characterized by a minimum inter-packet arrival time that is independent of packet size.

The optimality of PEDF when the criterion of minimizing the maximum lateness of packets was established in [8], however the relationship between PEDF and NPEDF in this context is new. The lexicographic optimality of PEDF is new as well.

Tracking policies have been proposed and studied in the context of Generalized Processor Sharing in [14, 7]. Theorems 1 and 2 appear in [14] but have been extended in this paper to include all tracking policies that obey a specific Ordering Property. The T(PEDF) and T(PEDF*) policies are new as are all of the results pertaining to these policies.

In the study of buffer-optimality, our results on Semi-Flexible allocation are new. The result on flexible allocation is straightforward, while the fixed allocation method has been studied before in [16],[4],[2],[9] [3] and the buffer optimal policy for non-zero σ_i presented in [15]. However, our result linking the schedulable region and buffer requirements under fixed allocation is new.

2 Tracking Service Disciplines

Given a pre-emptive policy π we derive a work-conserving, non-preemptive policy $T(\pi)$ that operates as follows: Let $f_p^\pi(t)$ be the time at which packet p departs from a multiplexer that implements policy π assuming that there are no arrivals after time t. Then at each decision epoch t of $T(\pi)$, the server schedules a packet that achieves the minimum value of $f_p^\pi(t)$ over all eligible packets present in the system at time t. Thus, $T(\pi)$ attempts to preserve the order in which packets depart under the pre-emptive system. At each decision epoch t the $T(\pi)$ server picks the next packet that would depart from the system under the pre-emptive system if no more packets were to arrive after time t. Since more than one packet may leave the preemptive system simultaneously, ties are broken arbitrarily.[2]

When π obeys the following *Ordering Property* we can establish a tight coupling between the sample paths of π and $T(\pi)$:

> Let packets p and p' both be in the system at time τ and suppose that packet p completes service before packet p' if there are no arrivals after time τ. Then packet p will also complete service before packet p' for any pattern of arrivals after time τ. Further, if p and p' leave the system simultaneously when there are no arrivals after time τ, then they leave system simultaneously for any pattern of arrivals after time τ.

A consequence of the ordering property is that if the tracking server schedules a packet p at time τ before another packet p' that is also backlogged at time τ, then packet p cannot leave later than packet p' in the preemptive system.

This leads to the following results (first developed in the context of Generalized Processor Sharing in [14]). Let f_p be the time at which packet p departs from the pre-emptive system and let $\hat{f}_p$ be the time it departs from the tracking system. Then:

Theorem 1 *Suppose the ordering property holds for the pre-emptive system and that the tracking service discipline is work conserving. For all packets p,*

$$\hat{f}_p - f_p < \frac{L_{\max}}{r}. \tag{6}$$

---

[2] Conceptually, a tracking policy simulates the performance of the pre-emptive system, but this can sometimes be accomplished without much computational overhead (see [14],[7]).

Theorem 2 *Suppose the ordering property holds for the pre-emptive policy* π: *Then for all times* $t \geq 0$ *and for each class* i:

$$Q_i^{T(\pi)}(t) - Q_i^{\pi}(t) \leq L_{\max}. \tag{7}$$

Examples of preemptive policies that obey the Ordering property include PEDF and head of the line Generalized Processor Sharing. Theorems 1 and 2 enable us to prove the delay and buffer optimality of various tracking service disciplines.

It is important to note that for many preemptive policies, there may be many ways to derive non-preemptive versions, and that these versions may differ significantly. For example consider the Preemptive Earliest Deadline First (PEDF) policy. Under the *oblivious* non-preemptive version of this policy (NPEDF), the next packet to be scheduled is always the one with the smallest deadline. The following example shows that order in which packets are scheduled under NPEDF may be quite different from that of T(PEDF):

Example. Assume that the server works at rate 1 and let packets arrive as follows: At time 0 a maximum size packet with deadline $ML_{\max}$ arrives and at time 0^+, K ($< L_{\max}$) 1-bit packets with deadline $ML_{\max}^-$ arrive. Thereafter, at each time $t = L_{\max}, 2L_{\max}, \ldots, ML_{\max}$, a maximum size packet with deadline $t + L_{\max}$ arrives. Under PEDF, the K 1-bit packets will be transmitted upon arrival and will depart from the system by time K and under both NPEDF and T(PEDF*) the packet arriving at time 0 must go into service upon arrival and departs at time $L_{\max}$. The difference in the two policies manifests at time $L_{\max}^+$, when T(PEDF*) serves the K 1-bit packets, while NPEDF serves the packet with deadline $2L_{max}$. Observe that in fact, the all 1-bit packets leave the system after time $(M-1)L_{max}$. This example will used again in Section 5. □

The example notwithstanding, the next result establishes that when the packet sizes are fixed, both T(PEDF) and NPEDF behave identically:

Lemma 1 *T(PEDF) behaves identically to NPEDF when all of the packet sizes are fixed at* $L \leq L_{\max}$..

3 Delay Optimal Policies

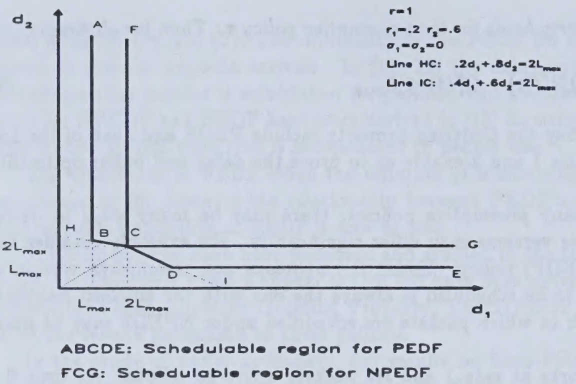
Let the non-negative vector $D = (D_1, ..., D_N)$ be a list of required (desired) upper bounds on delay so that no class i packet is delayed by more than D_i time units in the multiplexer. Thus D is schedulable under a policy π if and only if each packet p that arrives at time a_p and that belongs to class i is transmitted by time $a_p + D_i$. Packet p is said to have the *deadline* $d_p = a_p + D_i$. We now show the delay optimality of NPEDF and T(PEDF):

Theorem 3 *The NPEDF and T(PEDF) policies are delay-optimal among the class of non-preemptive policies. The schedulable regions of NPEDF and T(PEDF), are nonempty only if* $\sum_{i=1}^{N} \rho_i \leq r$, *and consists of the set of vectors which satisfy the constraints*

$$\begin{array}{rcl} (k+1)L_{\max} + \sum_{n=1}^{k} \sigma_{i_n} & \leq & D_{i_k}\left(r - \sum_{n=1}^{k-1} \rho_{i_n}\right) + \sum_{n=1}^{k-1} \rho_{i_n} D_{i_n}, \quad 1 \leq n \leq N-1 \\ N L_{\max} + \sum_{n=1}^{N} \sigma_{i_n} & \leq & D_{i_N}\left(r - \sum_{n=1}^{N-1} \rho_{i_n}\right) + \sum_{n=1}^{N-1} \rho_{i_n} D_{i_n}, \end{array}$$

whenever $D_{i_1} \leq D_{i_2} \leq \ldots \leq D_{i_N}$.

The lateness of a packet is defined as the difference of the packets' finishing time and its deadline. It is well known that under any arrival pattern the preemptive EDF policy (PEDF) minimizes the maximum lateness of all the packets. It follows that the PEDF policy is delay-optimal among all scheduling policies. The schedulable region of PEDF under the arrival patterns considered in this section can be found using similar arguments as those used to prove Theorem 3 and is given in the next Theorem.



d_2
A
F
r=1
r_1=.2 r_2=.6
σ_1=σ_2=0
Line HC: .2d_1+.8d_2=2L_{max}
Line IC: .4d_1+.6d_2=2L_{max}
H
B
C
$2L_{max}$
G
L_{max}
D
E
I
L_{max}
$2L_{max}$
d_1
ABCDE: Schedulable region for PEDF
FCG: Schedulable region for NPEDF

Figure 2: Schedulable regions for $\sigma_1 = \sigma_1 = 0$.

Theorem 4 *The schedulable region of PEDF is nonempty only if* $\sum_{i=1}^{N} \rho_i \leq r$, *and consists of the set of vectors which satisfy the constraints*

$$kL_{\max} + \sum_{n=1}^{k} \sigma_{i_n} \leq D_k \left(r - \sum_{n=1}^{k-1} \rho_{i_n} \right) + \sum_{i=1}^{k-1} \rho_{i_n} D_{i_n}, \ 1 \leq k \leq N,$$

whenever $D_{i_1} \leq D_{i_2} \leq \ldots \leq D_{i_N}$.

In Figures 2 and 3 we show the schedulable regions of PEDF and NPEDF under various parameters. As we see, in both figures the two regions differ by two strips which have width $L_{\max}/r$. In fact, by examining the schedulable regions it is easy to see that if the vector $\{D_1, \ldots, D_N\}$ is schedulable under PEDF, then the vector $\{D_1 + L_{\max}/r, \ldots, D_N + L_{\max}/r\}$ is schedulable under NPEDF. As we will see in the next section, this is a consequence of a general result that holds for any arrival patterns. Also, we see in Figure 2, where $\sigma_1 = \sigma_2 = 0$, that any schedulable vector under NPEDF has coordinates larger than $2L_{\max}/r$. Since the vector $\{2L_{\max}/r,\ 2L_{\max}/r\}$ is schedulable under the First-Come-First-Served (FCFS) policy, it follows that in this case from the point of schedulability there is no point in employing other scheduling policies. In fact, as can be seen from Theorem 3 this is true always when $N = 2$ and $\sigma_1 = \sigma_2 = 0$.

4 Minimization of Maximum Lateness

This criterion is of interest in situations where the deadlines represent a desirable time by which the packets should be transmitted and it is important to transmit each packet as early as possible and in a fair manner relative to the transmission times of the rest of the packets. It is well known that PEDF is a good policy with respect to this type of objectives in the sense that among all scheduling policies, it minimizes the maximum lateness of all packets under any arrival pattern. However, it is easy to construct arrival patterns for which the NPEDF is not optimal with respect to the min-max criterion among the non-preemptive policies. In spite of this, we will show that NPEDF is still a good policy in the sense that the maximum lateness under NPEDF is at most $L_{\max}/r$ larger than the maximum lateness under PEDF under any arrival pattern.

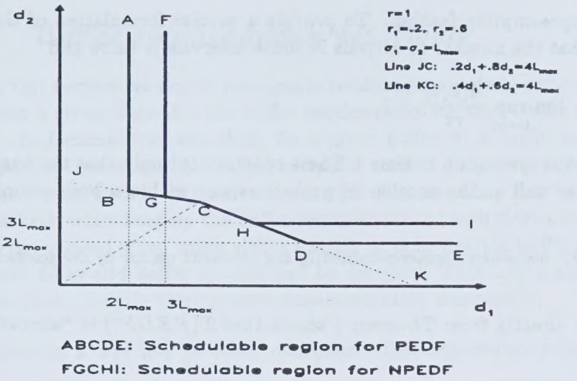

d_2
A
F
r=1
r_1=.2, r_2=.8
σ_1=σ_2=L_{max}
Line JC: .2d_1+.8d_2=4L_{max}
Line KC: .4d_1+.6d_2=4L_{max}
J
B
G
C
3L_{max}
2L_{max}
H
I
D
E
K
2L_{max}
3L_{max}
d_1
ABCDE: Schedulable region for PEDF
FGCHI: Schedulable region for NPEDF

Figure 3: Schedulable regions for $\sigma_1 + \sigma_1 \neq 0$.

Theorem 5 *Let f_p, $\hat{f}_p$ be the finishing time of packet p under the PEDF and NPEDF policies respectively and let d_p be its deadline. Then the following inequality holds under any arrival pattern:*

$$\sup_p \left\{ \hat{f}_p - d_p \right\} \leq \sup_p \left\{ f_p - d_p \right\} + \frac{L_{\max}}{r}.$$

Note from Theorem 1, that T(PEDF) has the following stronger property:

Theorem 6 *Let f_p, $\hat{f}_p$ be the finishing time of packet p under the PEDF and T(PEDF) policies respectively and let d_p be its deadline. Then, the following inequality holds for each packet p under any arrival pattern*

$$\hat{f}_p - d_p \leq f_p - d_p + \frac{L_{\max}}{r}.$$

5 Lexicographic Optimization.

A stronger optimality criterion than minimizing the maximum lateness which relates closer to fairness is the criterion of lexicographic optimization of packet lateness, which is defined below.

Let $\{l_i\}_{i=1}^n$, $\{u_i\}_{i=1}^n$, be two n-dimensional vectors and let $\pi_l(i)$, $\pi_u(i)$, be index permutations such that

$$l_{\pi_l(1)} \geq \ldots \geq l_{\pi_l(n)}, \qquad u_{\pi_u(1)} \geq \ldots \geq u_{\pi_u(n)}.$$

Vector $\{l_i\}_{i=1}^n$ is called lexicographically smaller than vector $\{u_i\}_{i=1}^n$, denoted as $\{l_i\}_{i=1}^n \leq_{lex} \{u_i\}_{i=1}^n$, if 1) $l_{\pi_l(1)} \leq u_{\pi_u(1)}$ and 2) $l_{\pi_l(i)} > u_{\pi_u(i)}$ for some $i = 2, \ldots, n$ implies that $l_{\pi_l(j)} < u_{\pi_u(j)}$ for some $j < i$. Let $\mathcal{A}$ be a set of n-dimensional vectors. Vector $\{l_i^*\}_{i=1}^n \in \mathcal{A}$ is lexicographically optimal in $\mathcal{A}$ if $\{l_i^*\}_{i=1}^n \leq_{lex} \{u_i\}_{i=1}^n$ for all $\{u_i\}_{i=1}^n \in \mathcal{A}$.

The property of the lexicographically optimal vector that relates to fairness is that if one attempts to reduce coordinate i by picking another vector in $\mathcal{A}$, then necessarily another coordinate that is larger than coordinate i will have to be increased (see [1, Section 6.5.2]).

It turns out that if preemptions are allowed, one of the *PEDF* policies is lexicographically optimal. Specifically, let *PEDF** be the policy that serves preemptively the packets with the earliest deadline first and among the packets with the earliest deadline it serves the packets with shortest remaining service time first. Among packets with the same deadline and the same remaining processing time,

$PEDF^*$ serves in an arbitrary but nonpreemptive fashion. To provide a precise formulation of the optimality of $PEDF^*$ we will assume that the number of arrivals in finite intervals is finite and

$$\limsup_{t\to\infty} \frac{A(0,t)}{rt} < 1,$$

where $A(0,t)$ is the work that arrives to the system up to time t. These constraints imply that the busy periods of any work-conserving policy, as well as the number of packets served within a busy period are finite.

Theorem 7 *Among all policies, $PEDF^*$ minimizes lexicographically the lateness vector of the packets that arrive during any busy period.*

The next observation, which follows directly from Theorem 1 shows that $T(PEDF^*)$ is "almost" lexicographically optimal.

Corollary 1 *Let $\{t_1\}_{p=1}^n$ and $\{l_1\}_{p=1}^n$ be the lateness vectors for a given set of arrivals when the service policy is T(PEDF) and PEDF respectively. Then for each packet arrival p:*

$$l_p \le t_p - L_{\max}.$$

The natural question to ask at this point is the performance of NPEDF with respect to the lexicographic criterion. The following example shows that for certain arrival patterns the policy is far from being lexicographically optimal.

Example. Consider the same arrival stream as in the example of Section 2. Since the K 1-bit packets will will be transmitted upon arrival by $PEDF^*$, therefore their lateness is $-M$. However, under any of the NPEDF policies, the lateness of the K packets is 0 while the lateness of the other packets remains essentially the same as when the $PEDF^*$ policy is employed. In other words, $PEDF^*$ improves the lateness of K packets by M, and yet only minimially penalizes the other packets. □

The example notwithstanding, NPEDF *is* almost lexicographically optimal for fixed size packets, since from Lemma 1, it behaves identically to T(PEDF).

6 Buffer-Optimal Policies

In this section we determine $B_{\min}$ (as defined in (5)) for flexible, semi-flexible and fixed buffer allocated multiplexers, and also give scheduling policies that achieve these values of $B_{\min}$.

Lemma 2 *For Flexible allocation $B_{\min} = NL_{\max} + \sum_{i=1}^N \sigma_i$, and this value is achieved by any work-conserving service policy.*

Lemma 3 *For Semi-Flexible allocation $B_{\min} = L_{\max}(2N-1) + \sum_{i=1}^N \sigma_i$, and this value is achieved by FCFS when $\sigma_i = 0$ for $i = 1, 2, ..., N$.*

Before dealing with the Fixed allocation case, we present a pre-emptive service policy called *Rate Proportional Processor Sharing (RPPS)* that was introduced in [15]. Recall that under our model, bits of a packet p are only eligible for service once the last bit of packet p has arrived. Let a class be backlogged at time t if a positive amount of eligible class i traffic is queued at time t. Then the RPPS server ensures that for any class i, if class i is continuously backlogged in the interval $[\tau, t]$ then

$$\frac{S_i(\tau,t)}{S_j(\tau,t)} \ge \frac{\rho_i}{\rho_j}, \quad j = 1, 2, ..., N. \tag{8}$$

Notice that if i and j are both continuously backlogged in the interval then (8) is met with equality. Also note that the RPPS obeys the Ordering Property discussed in Section 2. The following result is adapted from [15].

Lemma 4 *For Fixed allocation $B_{\min} = 2NL_{\max} + \sum_{i=1}^N \sigma_i$, and this value is achieved by T(RPPS).*

7 Buffer requirements v/s Delay

In this section we derive two simple relationships between the maximum delay experienced by packets from a given class and the buffer requirements for that traffic class.

In Lemma 5 we establish, for a given policy π, a useful lower bound on D_i^π as a function of M_i^π and the characteristics (σ_i, ρ_i) of traffic class i. Lemma 6 shows, that under a fixed buffer allocation there exist choices of traffic classes such that no policy can be designed that ensures low delays to all packets, while keeping the buffer requirements of each class small. As mentioned earlier, this limitation is not present when using either flexible or semi-flexible buffer allocations, for which policies which are both delay and buffer optimal can be designed. This may make such memory allocation schemes more desirable, despite their greater implementation complexity.

Lemma 5 *For any zero-loss multiplexer that implements policy π, we have:*

$$D_i^\pi \geq \frac{M_i^\pi - \sigma_i}{\rho_i} - L_{\max}\left(\frac{1}{\rho_i} - \frac{1}{r}\right) \tag{9}$$

The above result states that if a certain policy π limits the buffer requirements of traffic class i to M_i^π, then the maximum delay for that traffic class cannot be lower than a certain value function of both M_i^π and the characteristics of traffic class i. In the next Lemma, we strengthen this result in the more restricted case of fixed buffer allocation, for which we show that no policy can be "good" for both delay and buffer requirements.

Lemma 6 *If under a fixed buffer allocation scheme it holds that the buffer allocation b for each link is such that $b < NL_{max} + min_i \sigma_i$, then there is a choice of traffic classes $(\rho_1, \sigma_1), \ldots, (\rho_N, \sigma_N))$ consistent with (2) such that for any policy $\max_j D_j^\pi > d$ for any given $d > 0$.*

Corollary 2 *In a fixed buffer allocation multiplexer in which the service policy π guarantees bounded delay to all packets, independent of the choice of traffic class characteristics (condition (2) is obeyed): $B^\pi \geq N(NL_{\max} + \min_i \sigma_i)$.*

The above result means that the price paid for the simplicity of the fixed memory allocation scheme, will often be the need for significantly larger memory sizes than may be achievable with more flexible schemes.

8 Conclusions and Extensions

The definitions and results of this paper establish a framework within which various service policies can be evaluated and compared. We also proposed and analyzed a large class of policies called tracking policies, that may be of independent interest. Corollary 2 shows that under fixed allocation, delay-optimal policies cannot be buffer-optimal. We have recently established and are currently refining, a method to design delay-optimal policies that are *also* buffer-optimal under semi-flexible buffer allocation. This method will be included in a future version of this paper.

The natural direction in which our results should be extended is to multiple links, and this is our focus for future work.

References

[1] D. Bertsekas and R. Gallager. *Data Networks*. Prentice Hall, second edition, 1992.

[2] A. Birman, P. C. Chang, J. S.-C. Chen, and R. Guérin. Buffer sizing in an ISDN frame relay switch. Technical Report RC 14386, IBM Research, IBM T. J. Watson Research Center, P.O. Box 704, Yorktown Heights, NY 10598, August 1989.

[3] A. Birman, H. R. Gail, S. L. Hantler, Z. Rosberg, and M. Sidi. An optimal service policy for buffer systems. Technical Report RC 16641, IBM Research, IBM T. J. Watson Research Center, P.O. Box 704, Yorktown Heights, NY 10598, April 1991.

[4] I. Cidon, I. Gopal, G. Grover, and M. Sidi. Real-time packet switching: A performance analysis. *IEEE J. Sel. Areas Commun.*, SAC-6(9):1576–1586, December 1988.

[5] Rene L. Cruz. A calculus for network delay, Part I: Network elements in isolation. *IEEE Transactions on Information Theory*, 37(1):114–131, January 1991.

[6] Rene L. Cruz. A calculus for network delay, Part II: Network analysis. *IEEE Transactions on Information Theory*, 37(1):132–141, January 1991.

[7] Alan Demers, Srinivasan Keshav, and Scott Shenkar. Analysis and simulation of a fair queueing algorithm. *Internetworking Research and Experience*, 1(1), 1990.

[8] M. L. Dertouzos and A. K.-L. Mok. Multiprocessor on-line scheduling of hard-real-time tasks. *IEEE Trans. Softw. Eng.*, 15(12):1497–1506, December 1992.

[9] H. R. Gail, G. Grover, R. Guérin, S. L. Hantler Z. Rosberg, and M. Sidi. Buffer size requirements under longest queue first. *Performance Evaluation.* To appear. See also Proc. Perf. Dist. Syst. Integr. Commun. Sys., 1992.

[10] S. J. Golestani. Duration-limited statistical multiplexing of delay sensitive traffic in packet networks. In *Proceedings of IEEE INFOCOM '91*, 1991.

[11] K. Jeffay and R. Anderson. On optimal scheduling of periodic and sporadic tasks. Technical Report TR 88–11–06, University of Washington, University of Washington, Dept. Comput. Science FR-35, Seattle, WA 98195, November 1988.

[12] K. Jeffay, D. F. Stanat, and C. U. Martel. On non-preemptive scheduling of periodic and sporadic tasks. In *Proc. Real-Time Systems Symposium*, pages 129–139, San Antonio, TX, 1991. IEEE.

[13] C. L. Liu and J. W. Layland. Scheduling algorithms for multiprogramming in a hard-real-time environment. *Journal of the Association for Computing Machinery*, 20(1):46–61, 1973.

[14] A. K. Parekh. *A Generalized Processor Sharing Aproach to Flow Control in Integrated Services Networks.* PhD thesis, Department of Electrical Engineering and Computer Science, MIT, February 1992.

[15] A. K. Parekh and R. G. Gallager. A Generalized Processor Sharing approach to flow control in Integrated Services Networks—The Single Node Case. *ACM/IEEE Transactions on Networks*, 1(3), June 1993.

[16] G. Sasaki. Input buffer requirements for round robin polling systems. In *Proc. Allerton Conference on Communication, Control and Computing*, 1989.

[17] Lixia Zhang. *A New Architecture for Packet Switching Network Protocols.* PhD thesis, Department of Electrical Engineering and Computer Science, MIT, August 1989.

[18] Qin Zheng. *Real-time Fault-tolerant Communication in Computer Networks.* PhD thesis, University of Michigan, 1993.

ROBUST ROUTING FOR MAXIMUM WORST-CASE EXPECTED THROUGHPUT IN UNRELIABLE NETWORKS

Richard G. Ogier *Vlad Rutenburg*

SRI International
333 Ravenswood Ave.
Menlo Park, CA 94025

Abstract

This paper considers the problem of finding a multipath routing strategy that maximizes the minimum expected throughput from a source to a destination, where the minimum is taken over a class of random interference strategies in which a fixed amount of "energy" is divided among the network links. The problem is formulated as a two-person game, and, under certain assumptions, efficient solutions are presented for finding optimal routing and interference strategies. The general problem is shown to be NP-hard.

1 Introduction

This paper considers the important problem of robust routing in unreliable networks whose links can fail or deteriorate unpredictably, due to benign or hostile interference. More specifically, we consider the problem of finding a multipath routing strategy that maximizes the minimum expected throughput from a source to a destination, where the minimum is taken over a class of random interference strategies in which a fixed amount of "energy" is divided among the network links.

The problem is formulated as a two-person game, and, under certain assumptions, efficient solutions are presented for finding optimal routing and interference strategies. In particular, under these assumptions, we show that an optimal routing strategy is a maximum flow in a network with scaled link capacities, and that an optimal interference strategy is one that jams each link of a minimum cut with a certain probability. The general problem is shown to be NP-hard.

The authors recently presented a paper [1] that considers the problem of finding a routing strategy that minimizes the worst-case expected delay from every source to a single destination in an unreliable network, given a constraint on the number of outgoing links at each node that can be inoperable at any point in time. That paper also formulates the problem as a two-person game, and presents an efficient solution.

The paper is organized as follows. Section 2 formulates the general problem and shows that it is NP-hard. Section 3 considers the special case of unit link capacities, and Section 4 considers a modified problem with general link capacities.

2 Problem Formulation

We consider a directed graph $G = (V, E)$ with a single source node s and a single destination node d. Each link $l \in E$ has a capacity c_l and a jamming cost u_l. The jamming cost represents the amount of "energy" required to reduce the link's capacity to zero.

A routing strategy is represented by a flow $f = (f_p : p \in P)$, where P is the set of all paths from s to d and f_p is the flow of path p. We define the flow F_l of link l by

$$F_l = \sum_{p:\, l \in p} f_p \,. \tag{1}$$

A routing strategy must satisfy the following constraints:

$$\begin{aligned} f_p &\geq 0 \quad \text{for all } p & (2)\\ F_l &\leq c_l \quad \text{for all } l & (3)\\ \sum_p f_p &= R \,, & (4) \end{aligned}$$

where R is positive number representing the desired end-to-end throughput. A flow f satisfying these constraints is said to be *feasible.* We assume that a feasible flow exists. By the max-flow min-cut theorem [2], a feasible flow exists if and only if $R < C$, where C is the capacity of a min s-d cut.

An interference strategy is represented by a vector $x = (x_l : l \in E)$, where $0 \leq x_l \leq 1$. The number x_l represents the ratio of the jamming energy actually applied to link l, to the energy u_l required to completely disable the link. An interference strategy must satisfy the following constraints:

$$\begin{aligned} 0 \leq \ x_l \ &\leq 1 & (5)\\ \sum_l x_l u_l \ &\leq \ B \,, & (6) \end{aligned}$$

where B is a positive number representing the total energy. Thus, an interference strategy specifies how the total energy is divided among the network links. An interference vector x satisfying the above constraints is said to be *feasible.*

We let $S(f, x)$ denote the throughput achieved when the routing strategy is f and the capacity reduction vector is x. We assume that $S(f, x)$ has the form

$$S(f, x) = \sum_{p \in P} S_p(f, x), \tag{7}$$

where

$$\begin{aligned} S_p(f, x) &= f_p \text{ if } x_l = 0 \text{ for all } l \in p,\\ S_p(f, x) &= 0 \text{ if } x_l = 1 \text{ for some } l \in p \,. \end{aligned}$$

These two properties state that all of the traffic assigned to a path reaches the destination if no link of the path suffers interference, and that none of the traffic assigned to a path reaches the destination if any link of the path is completely disabled. We also assume that $S(f,x)$ is concave in f for each x and concave in x for each f. This is true, for example, if it holds for each function $S_p(f,x)$. Clearly, $S(f,x) = R$ if $x = 0$, and $S(f,x) = 0$ if $x_l = 1$ for all links l in some s-d cut.

To show that the concavity assumption is reasonable, we present two practical examples of $S_p(f,x)$ that satisfy this assumption. The first of these corresponds to a situation in which each packet transmitted on link l fails with probability x_l, and no packets are retransmitted:

$$S_p^1(f,x) = f_p \prod_{l \in p} (1 - x_l) . \tag{8}$$

The second example corresponds to a situation in which retransmissions are allowed, so that no reduction in throughput occurs until the effective link capacity $(1 - x_l)c_l$ is less than the link flow F_l:

$$S_p^2(f,x) = f_p \min_{l \in p} \left\{ 1, \frac{(1 - x_l)c_l}{F_l} \right\} . \tag{9}$$

It is easily verified that these functions satisfy the concavity assumption.

We consider a two-player game, called Game 1, in which Player 1 chooses the routing strategy f and Player 2 chooses the interference strategy x. Player 1 wishes to maximize the kernel $S(f,x)$ while Player 2 wishes to minimize it. Suppose $S(f,x)$ has a saddle point, i.e., a pair of strategies (f^*, x^*) such that, for all other strategies f and x,

$$S(f,x^*) \le S(f^*,x^*) \le S(f^*,x) . \tag{10}$$

Then the strategies f^* and x^* would be optimal for the game. However, $S(f,x)$ does not in general have a saddle point.

The strategies f and x defined above are *pure* strategies. In general, the game theoretic solution may involve *mixed* (i.e., random) strategies, so that Player 1's strategy is a probability distribution on the set of feasible flows f, and Player 2's strategy is a probability distribution on the set of feasible interference vectors x. Player 1 then tries to maximize the expected value of $S(f,x)$ (averaged over all strategies of both players) while Player 2 tries to minimize the expected value.

Since we assume $S(f,x)$ is concave in f for each x, and concave in x for each f, the theory of convex (or concave) games (e.g., [3]) implies that there exists an optimal strategy for Player 1 that is pure (not random), and that there exists an optimal strategy for Player 2 that is a mixture of strategies x that are vertices of the polytope of feasible interference vectors. We let $X = \sum_i q_i x^i$ denote a strategy for Player 2 that is a mixture of such vertices, where $\sum_i q_i = 1$, and q_i is the probability that vertex x^i is chosen. We let $S(f,X)$ denote the expected value of $S(f,x^i)$ over all pure strategies x^i used by X, that is,

$$S(f,X) = \sum_i q_i S(f,x^i) . \tag{11}$$

Thus, we wish to find strategies f^* and X^* such that, for any other pure strategy f for Player 1 and mixed strategy X for Player 2,

$$S(f,X^*) \le S(f^*,X^*) \le S(f^*,X) . \tag{12}$$

Such a pair is a solution to Game 1.

We next show that Game 1 is NP-hard, by showing that solving Game 1 implies solving the knapsack problem, which is known to be NP-complete [4].

Theorem 1 *Game 1 is NP-hard, and remains NP-hard even if $u_l = c_l$ for each link l.*

Proof Let L be a finite set, and for each element $l \in L$ let u_l and c_l be nonnegative integers denoting the size and value, respectively, of l. Let B be a positive integer. The knapsack problem is as follows: Select a subset L' of L that maximizes $\sum_{l\in L'} c_l$ subject to $\sum_{l\in L'} u_l \leq B$.

Now consider an instance of Game 1, where $V = \{s, d\}$ and there is a link l from s to d for each element $l \in L$. (Although this is a multigraph, it is easily modified to a simple graph by adding a node to the "middle" of each link.) For this game, let u_l, c_l, and B be the same as for the knapsack problem, and let $R = \sum_{l\in L} c_l$. We also assume that partial jamming has no effect, i.e., that $S_p(f, x) = f_p$ unless $x_l = 1$ for some link l in path p. (This allows us to restrict to 0-1 interference strategies.) Then clearly the optimal routing strategy is the one that saturates each link, i.e., $F_l = c_l$ for all l, and the optimal interference strategy is one that jams the set of links L' that maximizes $\sum_{l\in L'} c_l$ subject to $\sum_{l\in L'} u_l \leq B$. Therefore, the optimal interference strategy solves the knapsack problem, proving that Game 1 is NP-hard. Since the knapsack problem remains NP-hard even if $u_l = c_l$ for all l [4], so does the game. ■

3 Solution Assuming Unit Capacities and Costs

In this section, we assume that $c_l = u_l = 1$ for all links l, and that B is a positive integer, and we present an efficient solution to the resulting special case of Game 1, which we call Game 2. (We assume unit capacities and costs for convenience; we actually only require that the c_l be equal for all l, and that the u_l be equal for all l.)

Recall from Section 2 that the optimal routing strategy for Game 1 (and thus Game 2) is pure, and that the optimal interference strategy is a mixture of strategies x that are vertices of the polytope of feasible interference vectors. In the special case considered in this section, a vector x is a vertex of the polytope of feasible interference vectors if and only if $x_l = 1$ for exactly B links l and $x_l = 0$ for all other links. Therefore, the optimal interference strategy for Game 2 is a mixture of pure strategies, each of which fully jams exactly B links. We denote such a mixed strategy by $X = \sum_i q_i x^i$, where $\sum_i q_i = 1$, and q_i is the probability that the pure strategy x^i is used.

Let K be a minimum s-d cut, and let C denote the number of links in K. Clearly, Game 2 is interesting only if $B < C$, since otherwise the throughput is easily reduced to zero by jamming all links of K. We therefore assume $B < C$.

Let f^* denote the maximum flow in G from s to d, scaled by R/C. (Recall that we assume $R \leq C$.) Since the value of a maximum flow is C, the value of f^* is R, and so f^* is a feasible flow. We will show that f^* is the optimal routing strategy for Game 2.

Let $X^* = \sum_i q_i x^i$ be any mixed interference strategy that satisfies the following conditions:

A1. For each i, $x^i_l = 1$ for exactly B links l of the minimum cut K, and $x^i_l = 0$ for all other l.

A2. For each link $l \in K$,

$$\sum_i q_i x^i_l = \frac{B}{C}. \tag{13}$$

Condition A1 says that each pure interference strategy jams exactly B links of K, and Condition A2 says that each link in K is jammed with the same probability B/C. Any mixed strategy X^* satisfying conditions A1 and A2 is feasible, since $\sum_i x^i_l = B$ for each i. We will show that any such strategy X^* is optimal for Game 2.

An example of an interference strategy X^* satisfying A1 and A2 is the strategy that randomly selects a subset $G \subset K$ of size B, all such subsets having the same probability, and jams all links in G.

Another example is the following, where we assume the links of K are numbered from 1 to C. For $i = 1, ..., C$, let G_i denote the subset consisting of the B links of K beginning with link i, where we wrap around from link C to link 1. We let X^* be the interference strategy that randomly selects one of the C subsets G_i, each subset having the same probability $1/C$, and jams all links in G_i. This strategy clearly satisfies conditions A1 and A2.

Theorem 2 *Let f^* be the pure routing strategy defined above, and let X^* be any mixed interference strategy satisfying conditions A1 and A2. Then f^* and X^* satisfy Equation (12), i.e., they are the optimal strategies for Game 2.*

We postpone the proof of Theorem 2, since it will follow from the more general Theorem 3. Since a maximum flow and minimum cut can be computed in $O(|V|^3)$ time [2], Theorem 2 gives an algorithm for solving Game 2 in $O(|V|^3)$ time.

Intuitively, one can view the optimal routing strategy f^* as one that spreads the total flow R as evenly as possible over as many paths as possible. In fact, since f^* is the maximum flow scaled by R/C, the link flow F_l (corresponding to f^*) of any link l of any minimum s-d cut is equal to R/C, implying that the optimal routing strategy for Game 2 indeed spreads flow evenly over all links that belong to a minimum cut.

4 Solution Assuming Average Energy is Bounded

In this section, we wish to remove the constraint of unit link capacities that we assumed in the preceding section. However, Theorem 1 states that the problem is NP-hard for general link capacities, even if $c_l = u_l$ for all links l.

We therefore consider a modified problem, in which the constraint (6) that limits the energy of each pure strategy x^i of the mixed strategy $X = \sum_i q_i x^i$, is replaced by the following constraint, which limits the average energy over all pure strategies x^i belonging to the mixed strategy X:

$$\sum_i q_i \sum_l u_l x_l^i \ \leq \ B \ . \tag{14}$$

Letting $J_x = \sum_l u_l x_l$, and letting $\mathrm{E}[\cdot]$ denote the expectation with respect to q_i, the above equation can be written as $\mathrm{E}[J_X] \leq B$.

We thus consider the game, called Game 3, with kernel $S(f, X)$ defined by (11), where the routing strategy f satisfies (2)-(4), and the interference strategy $X = (q_i, x^i)$ satisfies $0 \leq x_l^i \leq 1$ and (14). Note that the interference strategies of Game 3 are already mixed. It is easy to verify that taking a mixture of these already mixed strategies does not introduce any new strategies.

Thus, the routing strategies f for Game 3 are the same as for Game 1. However, because the energy constraint (14) is less restrictive than the energy constraint (6) for Game 1, the set of interference strategies X for Game 3 is a superset of the set of mixed interference strategies for Game 1.

Since $S(f, X)$ is concave in f for each X, the theory of convex (or concave) games (e.g., [3]) implies that there exists an optimal strategy f^* for Game 3 that is pure, as was the case for Game 1. Therefore, a pair of strategies (f^*, X^*) is optimal for Game 3 if it satisfies

$$S(f, X^*) \leq S(f^*, X^*) \leq S(f^*, X) \tag{15}$$

for all strategies f and X of Game 3.

Let H denote the network G modified by letting the capacity of each link l be u_l instead of c_l. Let K be a minimum s-d cut in H, and let U denote the capacity of K in H, i.e., $U = \sum_{l \in K} u_l$. We assume that $B < U$, since otherwise the throughput is easily reduced to zero by jamming all links of K.

Let f^* denote a maximum flow in H from s to d, scaled by R/U. Since the value of a maximum flow is U, the value of f^* is R. (We do not assume that $R \leq U$, and therefore f^* may not be feasible for H.) We will show that if f^* is feasible for G, i.e., if the corresponding link flows F_l satisfy $F_l \leq c_l$ for all l, then f^* is optimal for Game 3. In particular, this will always be the case if $c_l = u_l$ for all l, since we assume that $R \leq C$, where C is the capacity of a minimum s-d cut in G.

By the max-flow min-cut theorem, the link flows F_l corresponding to f^* satisfy $F_l = u_l R/U$ for any link l in the minimum cut K. Thus, the optimal flows are distributed among the links of K in proportion to the amount of energy u_l required to completely jam link l.

Let $X^* = \sum_i q_i x^i$ be any interference strategy that satisfies the following conditions:

B1. For each i, x_l^i is 0 or 1 for each link l of K, and is 0 for each link l not in K.

B2. For each link $l \in K$,

$$\sum_i q_i x_l^i \ = \ \frac{B}{U} \ . \tag{16}$$

Condition B1 says that each strategy x^i is 0-1 and jams only links of the minimum cut K, and Condition B2 says that each link in K is jammed with the same probability B/U. Any mixed strategy X^* satisfying conditions B1 and B2 is feasible, since

$$\sum_i q_i \sum_l u_l x_l^i = \sum_{l \in K} u_l \sum_i q_i x_l^i = \sum_{l \in K} u_l B/U = UB/U = B \ , \tag{17}$$

that is, the average energy $\mathrm{E}[J_{X^*}]$ is equal to B.

It is easy to see that any strategy satisfying A1 and A2 also satisfies B1 and B2. Note also that the strategy X that jams each link of K independently with probability B/U satisfies B1 and B2.

The following theorem states that, if f^* is feasible for G, then f^* and X^* are optimal for Game 3.

Theorem 3 *Suppose that the flow f^* defined above is feasible for G, i.e., that the corresponding link flows F_l satisfy $F_l \leq c_l$ for all l. Then f^* and X^* satisfy (15), i.e., they are optimal strategies for Game 3. In particular, if $c_l = u_l$ for all l, then f^* and X^* are optimal.*

Proof We will use the notation $\Delta S(f, x) = S(f, 0) - S(f, x)$ and $\Delta_l S(f, x_l) = S(f, 0) - S(f, x')$, where $x'_l = x_l$ and $x'_m = 0$ for all $m \neq l$. Recall that $S(f, 0) = R$. Thus, $\Delta_l S(f, x_l)$ is the amount by which the throughput is reduced (compared to $x = 0$) if jamming energy $x_l u_l$ is applied to link l.

We first show that $S(f^*, X) \geq R - BR/U$ for all mixed strategies X satisfying the energy constraint (14), i.e., the constraint $\mathrm{E}[J_X] \leq B$.

Consider any pure strategy x, and consider a particular link l. Since $S(f, x)$ is concave for a fixed f, the ratio $\Delta_l S(f^*, x_l)/x_l$ is no greater than $\Delta_l S(f^*, 1)/1 = F_l$. Thus, $\Delta_l S(f^*, x)/(u_l x_l) \leq F_l/u_l$. Since f^* is a maximum flow in H scaled by R/U, we have $F_l \leq u_l R/U$. Therefore, $\Delta_l S(f^*, x_l)/(u_l x_l) \leq R/U$.

Recall that J_x denotes the total energy used by strategy x, i.e., $J_x = \sum_l u_l x_l$. We see that $\Delta S(f^*, x) \leq \sum_l \Delta_l S(f^*, x) \leq \sum_l u_l x_l R/U = J_x R/U$. Therefore, if $X = \sum q_i x^i$ is a mixed strategy satisfying $\mathrm{E}[J_X] \leq B$, then $\mathrm{E}[\Delta S(f^*, X)] \leq \mathrm{E}[J_X]R/U \leq BR/U$. Therefore, since $S(f^*, 0) = R$, we have $S(f^*, X)] \geq R - BR/U$.

We next show that $S(f^*, X^*) = R - BR/U$. Let x be some pure strategy in X^*. Recall that all x_l are either 0 or 1. By the max-flow min-cut theorem [2], $F_l = u_l R/U$ for all $l \in K$. Thus, if x is initially zero, setting $x_l = 1$ for any $l \in K$ reduces $S(f^*, x)$ by exactly $u_l R/U$. Since $F_l = 0$ for any link l that crosses K in the reverse direction, each path $p \in P$ with positive flow uses only one link of K. Therefore, it is easy to see that strategy x reduces $S(f^*, x)$ by exactly $\sum_l u_l x_l R/U = J_x R/U$. Averaging over all x in X^*, and recalling that $\mathrm{E}[J_{X^*}] = B$, we get the desired result that strategy X^* reduces $S(f^*, X)$ by exactly BR/U.

In summary, we have shown that

$$R - BR/U = S(f^*, X^*) \leq S(f^*, X) \text{ for all } X \ . \tag{18}$$

It remains to show that

$$S(f, X^*) \leq R - BR/U \text{ for all} f \ . \tag{19}$$

Let f be a feasible flow, and let X^* have the representation $\sum_i q_i x^i$. It suffices to show that

$$\sum_i q_i \Delta S(f, x^i) \geq BR/U \ . \tag{20}$$

Clearly,

$$\Delta S(f, x^i) = \sum_{p \in P(x^i)} f_p \ , \tag{21}$$

where $P(x^i)$ is the set of paths p that use at least one link that is jammed by x^i. Therefore

$$\sum_i q_i \Delta S(f, x^i) \ = \ \sum_i \sum_{p \in P(x^i)} q_i f_p \ = \ \sum_{p \in P} f_p \sum_{i : p \in P(x^i)} q_i \ . \tag{22}$$

Since K is a cut, every path $p \in P$ must contain at least one link of K. Let $l(p)$ denote any such link. Then

$$\sum_{p \in P} f_p \sum_{i : p \in P(x^i)} q_i \ \geq \ \sum_{p \in P} f_p \sum_i q_i x^i_{l(p)} \ = \ \sum_{p \in P} f_p B/U \ , \tag{23}$$

where the last equality follows from from condition B2. Since the total flow over all paths is equal to R, the last expression is equal to RB/U. ∎

We can now prove Theorem 2 as a special case of Theorem 3.

Proof of Theorem 2 Consider an instance of Game 2. Thus, $c_l = u_l = 1$ for all l. Let f^* and X^* be defined as in Theorem 2. Since f^* is feasible for Game 2, it is also feasible for Game 3. And since X^* satisfies conditions A1 and A2, it also satisfies the more general conditions B1 and B2. Therefore, Theorem 3 implies that f^* and X^* are optimal for Game 3. Since the set of feasible strategies for Game 2 is a subset of that for Game 3, it follows that f^* and X^* are also optimal for Game 2. ∎

References

[1] R.G. Ogier and V. Rutenburg. Robust routing for minimum worst-case expected delay in unreliable networks. In *Proc. IEEE INFOCOM*, 1993.

[2] E.L. Lawler. *Combinatorial Optimization: Networks and Matroids.* Holt, Rinehart, and Winston, 1976.

[3] S. Karlin. *Mathematical Methods and Theory in Games, Programming, and Economics.* Addison-Wesley, 1959.

[4] M.R. Garey and D.S. Johnson. *Computers and Intractibility – A Guide to the Theory of NP-Completeness.* W.H. Freeman and Company, 1979.

Adaptive Back-Presure Congestion Control Based on Local Information[1]

Leandros Tassiulas
Department of Electrical Engineering
Polytechnic University
6 Metrotech Center
Brooklyn, NY11201
leandros@olympos.poly.edu

Abstract

The problem of distributed congestion control as it arises in communication networks as well as manufacturing systems is studied in this paper. A multistage queueing system that models virtual circuit and datagram communication networks, and a class of manufacturing systems is considered. The topology may be arbitrary, there are multiple traffic classes, and the routing can be class dependent, with routes that may form direct or indirect loops. The model incorporates the functions of transmission scheduling, flow control and routing, through which congestion control is performed in the network. A policy is given that performs these functions jointly. According to the policy, heavily loaded queues are given higher priority in service. A congested node may reduce the flow from upstream nodes through a flow control mechanism. Whenever routing is required, it is performed such that the lightly loaded queues receive most of the traffic. For arrival processes with burstiness bounded by a deterministic bound, the policy guarranties bounded backlogs in the network, as long as the load of each server is less than one. The actions of each server are based on the state of its own queues and of the queues one hop away. Therefore they are implementable in a distributed fashion. An adaptive version of the policy is also provided that makes it independent of the arrival rates.

1. Introduction

Dynamic control achieves a better utilization of the transmission and switching resources of a communication network over static schemes. Dynamic schemes have been adopted for packet routing, for sharing the transmission capacity of a link among several competing information streams or for controlling the flow of traffic in a packet switching network. In large networks though, where the control functions are distributed, dynamic control policies are prone to misbehavior and the network may enter intriguing instability modes. The same phenomenon has been observed in manufacturing systems with distributed scheduling. In this work we present a new distributed dynamic control policy in a multistage queueing system that models virtual circuit and datagram

---

[1] *This research was supported in part by the Center for Advanced Technology in Telecommunications, Polytechnic University and by NSF under grant NCR-9211417*

communication networks as well as a class of manufacturing systems. The policy, that we call Adaptive Back-Presure (ABP) congestion control policy, schedules the servers, routes the served traffic and controls the flow based on local information and is therefore amenable to distributed implementation. Each server is allocated dynamically based on the state of the queues that it serves, and on the state of downstream queues, one hop away. When there are routing options, the decision is taken again in a similar distributed fashion. It is shown that under the ABP policy, the backlog in the network remains bounded as long as the utilization of each server is less than one.

A central problem in the design of Virtual Circuit (VC) communication networks is the sharing the transmission capacity of a link among the virtual circuits that go through it. Several schemes have been proposed including round robin, weighted round robin, golden ratio scheduling, virtual clock and processor sharing [Ha92, PPC92, Zh90, PaG92, Ka87]. In some of the above schemes [Ha91, Ka87] the link transmission scheduling is combined with window flow control. One of the main problems arising in this context is that of evaluating the throughput of a virtual circuit network, that is the session rates that can be sustained by the network for certain transmission control policies, without experiencing instabilities. Cruz [Cr91a, b] has considered that problem in virtual circuit networks employing several different link transmission disciplines including FIFO, LIFO and strict priority. He obtained bounds for the backlog in the nodes of the network. These bounds depend on the session arrival rates and burstiness characteristics and guarantee stable operation of the system for the traffic parameters for which they remain finite. In certain cases though, in networks with virtual circuits that may form cycles, these bounds explode for arrival rates which give utilization strictly less than 1 at all network links. In this case no conclusion can be drawn for the stability of the network. Chang [Ch92] extended these results obtaining backlog bounds in multiclass networks with routing. The stability problem though remained open for certain cases in networks with cycles. Yaron and Sidi [YaS93] addressed the same question with processes satisfying constraints on the tails of the backlog distribution. Hahne [Ha91] has shown that with round robin scheduling in each link and hop-by-hop window flow control, there exists window sizes to stabilize the network as long as the link utilization is less than one. The selection of the window sizes depends on the arrival rates. By the ABP policy, stabilization of the network is achieved as long as the utilization of every link is less than one and no knowledge of the arrival rates is required for its implementation. When the routes of the packets are not prespecified but only their final destinations are given, as in the case of datagram communication networks, then the ABP policy combines routing with the scheduling mechanisms described in the VC case, to preserve bounded queue lengths.

A problem of stability similar to the one we discussed above for VC networks, arises in manufacturing systems operated under distributed scheduling policies. Perkins and Kumar [PeK88] and Kumar and Seidman [KuS90] have studied the problem of stability in a flexible manufacturing system with distributed scheduling. While simple distributed policies are shown to stabilize acyclic manufacturing networks in [PeK88], a simple two stage queueing network is presented in [KuS90] where instabilities occur in situations at which all servers of the system are strictly underloaded and the system is operated under a work conserving policy. This example demonstrates how instabilities may occur in multistage manufacturing systems due to complicated phenomena of starvation and overloading. The queueing network that corresponds to the manufacturing system considered there, falls within the scope of the network we consider. The ABP policy stabilizes the manufacturing system as long as the utilization of each machine is less than one. The scheduling of each machine i is determined by the size of the backlog of the different part types in i as well as the size of the backlogs of those part types in downstream machines one hop away from i. Occasionaly the machines are forced to idle again depending on the local state. The queueing system considered here extends the above manufacturing systems model to include the case where a part type may have the option of several alternative manufacturing scenarios and routing decisions are made. In section 2.3 this is discussed in more detail. The ABP policy in that case combines scheduling and routing decisions to achieve the same goal.

The rest of the paper is organized as follows. In section 2 the queueing network is presented and the correspondence with virtual circuit and datagrams networks as well as manufacturing systems is demonstrated. In section 3 the issue of stability is discussed and the sufficient stability condition is given. In section 4 a parametric class of policies is specified and investigated. In section 5 the ABP policy is specified and investigated. Finally in section 6 the results are discussed and some open problems are presented. The proof of the results are not included and the interested reader is refered to [Ta93].

2. The network model

We consider a network consisting of N servers and B buffers. Each server i can serve any buffer from the set of buffers $\mathcal{B}_i$, $i = 1,..,N$; it is allocated to the buffers according to some scheduling discipline. The served traffic from buffer j can be directed to any buffer of the set $\mathcal{R}_j$. A routing policy determines to which buffer in $\mathcal{R}_j$ the traffic from j is routed to. From certain buffers the traffic can be directed out of the system; this is indicated by including a 0 in the set $\mathcal{R}_j$. We make the natural assumption that from every buffer the work can be forwarded out of the system if it is routed through an appropriate sequence of buffers. We consider a fluid model for the work comming into the system. The time is continuous and the instantaneous rate with which work comes to buffer i from outside is $a_i(t)$. For simplicity we assume that the arrival stream can contain no impulses, that is the arrival rate $a_i(t)$ for all buffers i is bounded uniformly over time by a bound λ. In the time interval (t_1, t_2) an amount of work equal to

$$\int_{t_1}^{t_2} a_i(t)dt$$

enters buffer i from outside. We assume that the arrival streams satisfy some burstiness constraints; that is there are nonnegative numbers a_i, b_i, $i = 1,..,B$ such that for all $0 \leq t_1 < t_2$ we have

$$\int_{t_1}^{t_2} a_i(t)dt \leq a_i(t_2 - t_1) + b_i.$$

It is conceptually simpler to think of the $a_i's$ as being the average arrival rates at the buffer. We will assume that the long-run average rate with which work enters the system in buffer i exists

$$\lim_{t\to\infty} \frac{1}{t} \int_0^t a_i(s)ds = a_i, \quad i = 1,..,B$$

and a_i will be reffered as the arrival rate to buffer i. This assumption is not needed for the validity of the results. When server i provides service to buffer $j \in \mathcal{B}_i$ which is non-empty, work leaves the buffer with a constant rate μ_{ij}, the *service rate* of server i

at buffer j. If server i is assigned to buffer j continuously for an amount of time T and the traffic is directed to buffer l in $\mathcal{R}_j$, then an amount of work $T\mu_{ij}$ is transferred from j to l. The selection of buffer $j \in \mathcal{B}_i$ served by i and of buffer $l \in \mathcal{R}_j$ where the traffic is directed to is done by the control policy. This work is routed to some buffer in $\mathcal{R}_j$. We allow more than one servers to serve the same buffer; in that case the service rate is assumed to be equal to the sum of the rates of the servers that serve the buffer. When server i that serves buffer j and directs the traffic to buffer k switches to buffers l and m respectively, a switchover time δ^i_{jklm} is involved during which the server idles. The results in the paper hold for arbitrary and distinct, from server to server, switchover times. For notational simplicity though and without loss of generality, we will assume that the switchover time is the same for all servers and buffers and it will be denoted by δ.

2.1 Virtual Circuit Networks

A class of networks where the above queueing model arises are the virtual circuit networks. A virtual circuit network is characterized by its topology graph $G = (V, E)$, the transmission rates of the links and the established virtual circuits $VC's$. The topology graph contains one node for every network node and one directed link $e = (v, w)$ for every communication link from node v to node w. The transmission rate of link e is denoted by μ_e. A virtual circuit i is specified by the sequence of links $e^i_1, .., e^i_N$ that traverses as it goes through the network. The traffic of VC i enters the network at the origin node of e^i_1 and leaves it at the destination node of link e^i_N. A scheduling policy schedules for each link the transmissions of the virtual circuits going through it. Also at certain times the link may be forced to idle due to flow control actions.

A virtual circuit network is modeled by the above queueing system as follows. Each link corresponds to a server with transmission capacity equal to its service rate. Clearly in this case the service rate of server (link) i is the same for all buffers in $\mathcal{B}_i$. There is one buffer for every VC i and every link e^i_k traversed by i. The buffers of virtual circuit i contain traffic of that VC waiting to be transmitted through the corresponding link. When a link is allocated to any of the buffers of the VC's going through it, that buffer empties with rate equal to the link transmission rate. The traffic from the buffer of virtual circuit i at link e^i_j is routed to the buffer of the same virtual circuit at link e^i_{j+1}, except if e^i_j is the last link traversed by VC i; in the latter case the traffic leaves the system . Hence the set $\mathcal{R}_j$ contains a single element for each j and no routing is needed. The traffic in the VC's consists of streams of packets and clearly the link cannot switch from VC to VC in a time period smaller than a packet transmission time. In the assumptions we did though in the previous section, the traffic is considered as a continuous flow. For the results that we obtain here it is not important whether the traffic is continuous or in terms of packets.

2.2 Datagram Networks

In datagram networks the traffic of a communication session does not have to follow a specific route but can be routed arbitrarily and different packets may follow different paths to the destination. The packets are differentiated only by their destination nodes. When a packet arrives at a node that is not its final destination, a decision is taken which outgoing link the packet will follow next; then the packet is placed at the outgoing buffer of that link. There are no constraints on which outgoing link the packets of each specific destination will follow; this is determined by the routing policy. Packets of several different destinations are waiting in the buffer of each outgoing link to be transmitted. The link scheduling discipline determines which is the packet to be transmitted next.

In the corresponding queueing model, the buffers are in one-to-one correspondence with the links. For each server i, there is one buffer in $\mathcal{B}_i$ for each destination node. The traffic with destination node k, that is transmitted through link i, is stored in the corresponding buffer of $\mathcal{B}_i$. These buffers do not necessarily correspond to physical buffers in the origin node of link i. Their introduction enables the differentiation of the traffic, based on the destination and the link the traffic will go through. If from the topology of the network or from other constraints, it turns out that traffic of a specific destination never transverses link i, then the corresponding buffer in $\mathcal{B}_i$ will be permanently empty.

2.3 Manufacturing systems

In manufacturing systems several different types of parts are fabricated. A part type, in order to be manufactured needs to be processed by a number of machines in some prespecified order. In flexible manufacturing systems a machine can process more than one type of parts. In this case the machine has to switch from one part type to another at certain times since the different parts are not processed simultaneously. The switching of the machine involves a period during which the machine idles. The switching should be done rare enough such that the fraction of time that the machine is utilized is higher than the loading.

Perkins and Kumar [PeK89] have obtained a distributed scheduling policy that stabilizes any acyclic network of manufacturing machines. When cycles are formed by the routes of the parts in the manufacturing system then the behavior of the system is more difficult to characterize. The intrinsic complexity of the problem was demonstrated by a counterintuitive example of instability in a simple manufacturing system in [KuS90]. The queueing network considered here is readily interpreted into a manufacturing system model. The servers correspond to the machines, the arrival streams to the different part types and the buffers of each machine store the part types at intermediate processing stages. If a part type needs to be processed by a machine more than once in its manufacturing cycle then it waits in a different buffer each time. When the part types have unique routes through the manufacturing system then the sets $\mathcal{R}_j$ have a unique element, the next buffer that the parts will join when they leave from j. Kumar and Seidman in [KuS90] propose a scheduling policy with a supervisor mechanism that stabilize the manufacturing system as long as the load of each machine is less than one. The supervisory mechanism though relies on knowledge of the arrival rates. The ABP policy stabilizes the system without the arrival rate information. The model considered here extends the one considered in [PeK89, KuS90] to include cases where a part type has the option to follow different routes at certain points of its processing in which case routing is performed.

3. Necessary and Sufficient Stabilizability Conditions

We are interested on the average rate with which work can be served. We consider that certain throughput rates can be achieved if there exist a policy under which the network is *stable* when the long-run average arrival rates are equal to the desired throughputs. Denote by $X_j(t)$ the amount of work in buffer j at the time t.

Definition: The system is *stable* under a policy π if

$$\limsup_{t\to\infty} \sum_{j=1}^{B} X_j(t) < D$$

where D may depend only on the arrival rates $\mathbf{a} = (a_1, .., a_N)$, the burstiness coefficients $\mathbf{b} = (b_1, .., b_N)$ and the service rates $\mu = (\mu_{ij} : i = 1, .., N, \ j \in \mathcal{B}_i)$ but not on the initial condition.

The following condition on the arrival and service rates is necessary and sufficient for the existence of a policy under which the network is stable.

C1. *There exist nonnegative numbers* $f_{jk} \quad j = 1, ..B, \ k \in R_j$ *that satisfy the flow conservation equations*

$$a_j + \sum_{k:j\in R_k} f_{kj} = \sum_{k\in R_j} f_{jk}, \ j = 1, .., B. \tag{3.1}$$

Also there exist nonnegative numbers u^i_j, $i = 1, .., N, \ j = 1, .., B$ *such that*

$$\sum_{j\in\mathcal{B}_i} u^i_j < 1, \quad i = 1, .., N, \tag{3.2a}$$

$$\sum_{l\in R_j} f_{jl} \leq \sum_{i:j\in\mathcal{B}_i} u^i_j \mu_{ij}, \quad j = 1, .., B. \tag{3.2b}$$

An intuitive justification of C1 follows. The number f_{jk} in the condition $C1$ represents the long-run average rate with which work is transferred from buffer j to buffer k. Hence (3.1) is indeed a flow conservation equation. A collection of nonegative numbers $f_{jk}, j = 1, .., B, k \in \mathbf{R}_j$ which satisfy equations (3.1) will be referred to as *flow* in the following. The number u^i_j represents the fraction of time that server i spends serving buffer j. Inequality (3.2) expresses the fact that the rate with which work leaves buffer j (the sum in the left side of (3.2)) cannot be greater than the server capacity (the sum in the right side of (3.2)) that is allocated to that buffer.

In the case of a virtual circuit network condition C1 is equivalent to the condition that for every link, the sum of the average rates of all virtual circuits that go through the link is strictly less than its transmission capacity. This can be easily seen by observing the following. Since there is no routing involved, $\mathcal{R}_j$ contains a single element for all $j's$ and f_{jk} is equal to the arrival rate of the virtual circuit to which buffer j corresponds to. Also each buffer can be served by one server (link) only therefore the sums at the second inequality in 3.2 are reduced to single terms; these inequalities are equivalent to the fact that the traffic load of every link is less than one.

3.1 Necessity of C1 for stability

The necessity follows when the arrival streams satisfy the following condition.

S1 There exist nonegative numbers $\hat{b}_i, \ i = 1, .., B$ such that

$$\int_{t_1}^{t_2} a_i(\tau)d\tau \geq a_i(t_2 - t_1) - \hat{b}_i$$

Condition $S1$ can be viewed as a constraint on the idling of the arrival streams.

Theorem 3.1: If the system is stable and condition $S1$ holds, then condition $C1$ holds as well.

The necessity of strict inequality in relation (3.2a) of condition $C1$ is due to the definition of stability that we have and more specifically to the requirement that the assymptotic bound of the backlog is idependent of the initial conditions. It is possible that bounded backlog is guarranted without strict inequality in (3.2a). This bound though cannot be independent of the initial condition. This illustarted in the following counterexample. Consider the case of a single server queue with constant instantaneous arrival rate equal to a and service rate μ. Condition $C1$, with strict inequality in (3.2a), boils down to the condition $a < \mu$. If equality is allowed in (3.2a) we may have $a = \mu$. In the latter case, and if the instantaneous arrival rate is constant and equal to the service rate μ then the backlog at all times will be equal to the backlog at the time instant 0. Therefore the network will be unstable according to the definition of stability considered in the paper, since there is no assymptotic bound in the backlog independent of the initial condition.

4. The Parametric Back-Pressure Policy

In this section we present and study the Parametric Back-pressure Policy PBP(α). This policy determines whether a server idles or not and if it does not idle, which queue is served and where the traffic is directed to. The frequency with which a server is switched from queue to queue depends on a parameter $\alpha > 1$. We include α as an argument in the name of the policy to emphasize that dependence.

$PBP(\alpha)$

While the server i serves a buffer j_0 and directs the traffic to a buffer k_0, it is constantly monitoring the quantity

$$A_i(t) = \max_{l \in \mathcal{B}_i} \{\mu_{il} \max_{n \in \mathcal{R}_l} \{X_l(t) - X_n(t)\}\} \tag{4.1}$$

If

$$\alpha \mu_{ij_0}(X_{j_0}(t) - X_{k_0}(t)) > 0$$

and

$$A_i(t) \geq \alpha \mu_{ij_0}(X_{j_0}(t) - X_{k_0}(t))$$

then the server is rescheduled to serve the queue $l \in \mathcal{B}_i$ and the traffic is directed to queue $n \in \mathcal{R}_l$ which together realize the maximum in (4.1), with ties broken arbitrarily. Service is reassumed after the switchover time δ.
If

$$A_i(t) \leq 0,$$

then the server idles for a time period δ. If at the end of the idling period it is still $A_i(t) \leq 0$, then the server restarts an idling period of the same duration. Otherwise server i is rescheduled to serve the queue $l \in \mathcal{B}_i$ and the traffic is directed to queue $n \in \mathcal{R}_l$ which together realize the maximum in (4.1).

According to $PBP(\alpha)$, the served traffic of buffer l is routed to buffer n which achieves the maximum in

$$\max_{n \in \mathcal{R}_l} \{X_l(t) - X_n(t)\} \tag{4.1a}$$

whenever buffer l is served. Server i selects which buffer $l \in \mathcal{B}_i$ to serve based on the terms

$$\mu_{il} \max_{n \in \mathcal{R}_l} \{X_l(t) - X_n(t)\}, \quad l \in \mathcal{B}_i.$$

It selects the one which achieves the maximum in (4.1). In order to avoid server i switching from queue to queue too often, the policy reschedules the server only if the quantity (4.1a) for the queue l under service becomes considerably smaller than $A_i(t)$.

How much smaller it should become is determined by α which regulates how often the server switches. This feature is reminiscent of the clear a fraction policy considered in [PeK89]. If the backlog in all the downstream queues is larger than the backlog in the queues of $\mathcal{B}_i$, that is the quantity $A_i(t)$ is negative, then the server i idles. Note that the scheduling of server i relies on information about the queue lengths of the buffers in $\mathcal{B}_i$ and in $\mathcal{R}_j$, $j \in \mathcal{B}_i$. This is local information for server i in several practical systems. For example in the case of the VC network, the buffers in $\mathcal{B}_i$ and in $\mathcal{R}_j$, $j \in \mathcal{B}_i$, reside in the origin and destination node of the server, therefore policy $PBP(\alpha)$ can be implemented in a distributed fashion.

Notice that the actions taken by policy $PBP(\alpha)$, in the general network they correspond to scheduling the server, routing of the traffic and idling of the server. If routing is not involved then the policy is simplified considerably. This is the case in a virtual circuit network where $PBP(\alpha)$ acts as follows. Every link is allocated to the virtual circuits that go through it based on the backlogs of the buffers of these VC's in its origin node and in its destination node. For each VC, the differences of the backlog of the buffer in the origin node of the link minus that of the buffer in the destination node of the link are formed. If all the differences are negative then the link idles. Otherwise the quantity

$$A_i(t) = \max_{l \in \mathcal{B}_i}\{X_l(t) - X_{\hat{l}}(t)\} \tag{4.1b}$$

is computed where l and $\hat{l}$ are the buffers that correspond to the same virtual circuit in the origin and destination node respectively of link i. If

$$A_i(t) > \alpha\mu_{ij}(X_j(t) - X_{\hat{j}}(t))$$

where j, $\hat{j}$ are the buffers that correspond to the VC currently transmitted, then the server switches to the VC that achieves the maximum in (4.1b).
The stabilizability properties of $PBP(\alpha)$ are stated in the following.

Theorem 4.1: If the arrivals and service rates satisfy condition $C1$ then there exists an $\alpha > 1$ such that the system is stable under $PBP(\alpha)$.

The proof of the theorem is based on the study of the drift of a quadratic Liapunov function.

The $PBP(\alpha)$ policy achieves a bounded backlog in the network under the necessary and sufficient stabilizability condition C1 and is indeed distributed since the decisions at each node rely on one hop away state information. Nevertheless the arrival and service rates and the topology of the network need to be known in order to select the appropriate α. In the next section the ABP policy, an adaptive version of the PBP, is obtained which does not rely on knowledge of the arrival and service rates. Note that in the model it is possible work to go through a buffer more than once. It appears that this possibility may lead to instabilities under a distributed policy. The following is an intuitive explanation of why this is not happening with $PBP(\alpha)$. The goal of the policy is to push all the traffic out of the network. It is possible that work may visit a buffer more than once, following a cycle of buffers with small backlogs, instead of following a route out of the network, when the buffers that lead out of the network are congested. Eventually though, the latter buffers will empty and the work will find its way out of the network by selecting the route with the low backlog buffers.

5. The Adaptive Back-Pressure Policy

The adaptive back-pressure policy is identical to $PBP(\alpha)$ except that α is not preselected but is computed at each node based on the local (one hop away) state. Each server i computes a parameter $\alpha_i(t)$ based on the lengths of the queues that it may serve and the queues that it may direct the traffic to, as follows. Consider a function $g : R^+ \to R^+$ which is nonincreasing, strictly greater than 1 and such that

$$\lim_{x\to\infty} x(g(x)-1) = \infty \tag{5.1}$$

The function $g(x) = 1 + x^{-a}$ satisfies (5.1) for all $a < 1$. Define $\alpha_i(t)$ as

$$\alpha_i(t) = g(\max_{j\in\mathcal{B}_i}\{\mu_{ij}\max_{l\in\mathcal{R}_j}\{X_j(t) - X_l(t)\}\})$$

The policy ABP is as follows

ABP

Each server acts as in the $PBP(\alpha)$ policy with the difference that server i uses the locally computed $\alpha_i(t)$, instead of α.

The following holds for ABP.

Theorem 5.1: The network is stable under ABP if condition $C1$ holds.

The proof of the theorem is based on the study of the drift of a quadratic Liapunov function.

6. Conclusions and discussion

In the ABP policy each server is scheduled based on the lengths of the queues that it serves and of the queues one hop away, which were assumed to be instaneously available to the server in our study. This is not always the case though. For example in the virtual circuit network the link scheduling was based on the queue lengths at the origin and the destination nodes of the link. Clearly the queue lengths in the destination node could not be made available at the origin node without a delay greater than or equal to the propagation delay of the link. In high speed networks the link propagation time is enough for the state of the queues to change considerably. Therefore there will be a discrepancy between the actual queue lengths and those available to the server. Nevertheless, as long as the difference between the actual queue lengths and those available to the server is bounded independently of the queue length values, then the results obtained in the paper remain unaffected. That is the ABP policy stabilizes the network with delayed information about the queue lengths as well.

According to the ABP policy, server i idles if the quantity $A_i(t)$ becomes less than or equal to 0 or in other words if for every queue $l \in \mathcal{B}_i$ the backlog of all queues in $\mathcal{R}_l$ is greater than or equal to that of queue l. The results remain unaffected when $A_i(t)$ is compared with an arbitrary negative number instead of 0 in the definition of ABP. That is the server idles only if the backlog in the down stream queues becomes greater than that of the upstream queues by a certain amount.

The delay through the network is an issue left open for further research. Bounded backlogs imply bounded delay, therefore under the ABP policy the delay will be finite when the stability condition C1 holds. There are several questions which are left open though. How does the delay varies when the function g that estimates the parameter α in each node changes? Which is a good choice of g as far as the delay is concerned? As we said earlier, the backlog in the network remains bounded even if the server i idles whenever $A_i(t) < h$ for an arbitrary h and not necessarily for $h = 0$. What is a good choice of h for small delays? Also, what will be the effect of the delayed information about the state of the one hop away queues on the delays through the network and how the latter is affected from the propagation delay? The investigation of these questions might lead to refinements of ABP with improved performance with respect to delay.

References

[Ch92] C.S. Chang, "Stability, queue length and delay, part i: Deterministic queueing networks," Technical Report RC 17708, IBM T.J. Watson Research Center, Yorktown Heights, NY 10598, 1992.

[Cr91a] R. L. Cruz, "A Calculus of Network Delay, Part I: Network Elements in Isolation," IEEE Trans. on Information Theory, IT-37, January 1991.

[Cr91b] R. L. Cruz, "A Calculus of Network Delay, Part II: Network Analysis," IEEE Trans. on Information Theory, IT-37, January 1991.

[Ha91] E.L. Hahne, "Round-Robin Scheduling for Max-Min Fairness in Data Networks," IEEE Journal on Sel. Areas in Commun., vol. JSAC-9, p. 1024-1039, September 1991.

[Ka87] M. G. H. Katevenis, "Fast switching and fair control of congested flow in broadband networks," IEEE J. Select. Areas Commun., vol. 5, no. 8, pp. 1315-1326, Oct. 1987.

[KS90] P.R. Kumar, T.I. Seidman, "Distributed Instabilities and Stabilization Methods in Distributed Real Time Scheduling of Manufacturing Systems," IEEE Trans. on Automat. Contr., AC-35, p. 289-298, 1989.

[PaG92] A.K. Parekh, R.G. Gallager, "A Generalized Processor Sharing Approach to Flow Control in Integrated Service Networks," INFOCOM'92.

[PPC92] S.S. Panwar, T.K. Philips, MS. Chen, "Golden Ratio Scheduling for Flow Control with Low Buffer Requirements," IEEE Trans. on Communications, vol. COM-40, p.765-772, April 1992.

[PeK89] J. Perkins, P.R. Kumar," Stable, Distributed, Real-Time Scheduling of Flexible Manufacturing Assembly/Disassembly Systems," IEEE Trans. on Automat. Contr., AC-34, pp.139-148, Feb.1989.

[Ta93] L.Tassiulas, "Adaptive Back-Pressure Congestion Control Based on Local Information," Technical Report CATT93-4, Polytechnic University, accepted for publication in the IEEE Transactions on Automatic Control.

[TaE92] L. Tassiulas, A. Ephremides, "Stability Properties of Constrained Queueing Systems and Scheduling for Maximum Throughput in Multihop Radio Networks," IEEE Transactions on Automatic Control, pp.1936-1948, December 1992.

[YaS93] O. Yaron and M. Sidi, "Calculating performance bounds in communication networks," IEEE INFOCOM, pp. 539–546, 1993.

[Zh90] L. Zhang, "Virtual Clock: A new traffic control algorithm for packet switched networks," SIGCOM'90, Philadelphia, PA, 1990.

STABILITY ANALYSIS OF SCHEDULING POLICIES IN RING NETWORKS WITH SPATIAL REUSE

Leonidas Georgiadis
IBM T. J. Watson Research Ctr.
P.O. Box 704
Yorktown Heights, NY 10598
U.S.A

Wojciech Szpankowski*
Dept. Computer Science
Purdue University
Computer Science Build.
W. Lafayette, IN 47907
U.S.A.

Leandros Tassiulas†
Dept. Electrical Engineering
Polytechnic University
6 Metrotech Center
Brooklyn, NY 11201
U.S.A.

1 Introduction

We consider a ring with spatial reuse, i.e., a ring in which multiple simultaneous transmissions are allowed as long as they take place over different links (cf. [4, 5, 6]. Time is divided in slots and each slot is equal to the smallest transmission unit, called packet. We assume zero propagation delay. A node can transmit a packet at the outgoing link at the same time that it receives another packet in the incoming link. A node receiving a packet with destination another node in the ring, may retransmit the packet in the outgoing link *in the same slot*, i.e., the ring has *cut-through* capabilities.

In [5], [4], a policy is proposed for the operation of the ring. Each node is assigned a number called "quota". The policy operates in cycles. A node is allowed to transmit during a cycle as long as the number of transmitted packets does not exceed its assigned quota. An analysis of the throughput characteristics of this policy is provided in [6], where it is also shown that that if the end-to-end throughput requirements result in aggregate traffic load for each link of the network less than one, then the quotas can be selected to achieve these throughput requirements. In order to compute the appropriate quota values, however, it is necessary to know the end-to-end transmission rates a priori.

In this paper we first propose an adaptive version of the quota policy which has maximal stability region. The adaptive policy does not need to preselect the quota based on the end-to-end traffic requirements. During the operation of the system, each node readjusts its quota based on the size of its queue. We denote such a policy as Π. More precisely, the proposed policy operates in cycles. At the beginning of a cycle each node allocates itself "quota" equal to the number of packets at its buffer. During a cycle a node can transmit no more messages than the quota allocated to it. A cycle ends when *all* nodes transmit their quota number of packets as well as all the packets that they may have received through their incoming link and are destined to another node. The proposed policy requires a distributed mechanism by which every node realizes that all the other nodes completed their quota and thus a cycle ends. Such a mechanism is provided in [5]. In [2], the stability of the ring was studied for stationary arrival processes and a policy was proposed that stabilizes the ring under the same conditions for the arrival rates. The analysis here is for a more restricted arrival process but the proposed policy is considerably simpler than the one proposed in [2].

---

*This research was primary done while the second author was visiting INRIA in Rocquencourt, France. The author wishes to thank INRIA (projects ALGO, MEVAL and REFLECS) for a generous support. Additional support was provided by NSF Grants NCR-9206315 and CCR-9201078 and INT-8912631, and from Grant AFOSR-90-0107, and in part by NATO Grant 0057/89.

†The research of the third author was supported in part by NSF under grant NCR-9211417 and by the New York State Center for Advanced Technology in Telecommunications, Polytechnic University.

We next provide the stability analysis of another policy proposed in [5], [4]. In this policy the quotas are preallocated to each node and do not change during the operation of the system. The policy ensures the fair access to the ring when the packet arrival rates to the nodes fluctuate and may even cause the system to operate in an unstable regime. In effect, the cycle lengths remain bounded irrespective of the load on the ring. In this way, the operation of a node with regular traffic requirements is not adversely affected by nodes that may become overloaded. This property is not shared by the adaptive policy. As we shall see, however, an implication of imposing a bound on the number of packets that may be transmitted by a station is that the stability region of the finite-quota policy is in general smaller than the stability region of the adaptive policy. The stability region in this case is very complicated and depends on the arrival rate distributions. After providing the general methodology of deriving the stability region, we provide bounds for the region that can be computed easier. For the case of two nodes, we provide closed-form expressions for the region of arrival rates under which the system is stable for any arrival distribution.

This paper is organized as follows. In the next section we describe the stability region for the adaptive policy, and show that it is maximal. In section 3 we provide the stability analysis of the policy that operates with fixed quota as well as computable upper and lower for the stability region. Finally, for the case of two nodes we provide the region of arrival rates under which the system is stable for any arrival distribution.

2 Stability of Ring with Adaptive Policy

Let M be the number of nodes and set $\mathcal{M} = \{1, \ldots, M\}$. The operations $i \oplus j$ and $i \ominus j$ denote respectively, addition and subtraction modulo M, with the convention that index 0 refers to node M. Furthermore, when i, j refer to node indices we denote $\sum_{k=i}^{j} x_k := x_i + x_{i\oplus 1} + \ldots + x_{j\ominus 1} + x_j$. We assume that the nodes are arranged on the ring according to their index so that the outgoing link to node i is the incoming link for node $i \oplus 1$. Node i may receive external traffic with destination any other node j in the system. Let $R_{ij}(t)$ be the number of packets that arrive at node i from the outside with destination node j. If $i = j$, then it is assumed that the packet has to cross all the nodes in the ring until it is received by the originating node, i.

The proposed policy Π operates in cycles and is based on the idea of allocating quotas to the nodes, proposed in [5],[4]. Let τ_k be the beginning of the kth cycle and set $\tau_1 = 1$. At time τ_k each node allocates itself "quota" $\nu_i(k) = Q_i(\tau_k)$, where $Q_i(t)$ is the queue size of node i at time t. Node i can transmit up to $\nu_i(k)$ packets during cycle k according to *any fixed nonidling* policy, i.e., the only restriction that is imposed on the transmissions is that the node puts a packet in its outgoing link whenever either its queue is nonempty, or a message is received in the same slot in its incoming link with destination another node on the ring. Cycle k ends when *all* nodes transmit $\nu_i(k)$ packets as well as all the packets that they may have received through their incoming link and are destined to another node.

2.1 Main Results

Throughout this section we adopt the following assumption:

(A1) The vector process $\{\mathbf{R}(t)\}_{t=1}^{\infty}$, where $\mathbf{R}(t) = \{R_{ij}(t),\ i, j \in \mathcal{M}\}$, consists of i.i.d. vectors. We denote $R_{ij} := R_{ij}(1)$. Note that we do not make any independence assumptions for the work arriving in various nodes at the same slot. To avoid technical difficulties we will also assume that $\Pr(R_{ij}(t) = 0,\ i, j \in \mathcal{M}) > 0$.

In order to formulate our main results in a compact form, we need some additional notation. Let

$$\rho_{ij} := \lim_{n\to\infty} \frac{\sum_{t=1}^{n} R_{ij}(t)}{n} = ER_{ij}\ . \tag{1}$$

We also define $\alpha_{im} := \sum_{m\oplus 1}^{i} \rho_{ij}$ and $r_m = \sum_{i\in\mathcal{M}} \alpha_{im}$. Note that α_{im} is the average number of packets per slot that are generated by node i and cross node m, and r_m is the average number of packets per slot, that cross node m. Finally, we set $r = \max\{r_m : m \in \mathcal{M}\}$.

It has been shown in [6] that for a fairly large class of policies for the ring, the throughput vector should lie within the following region $\mathcal{R} = \{\boldsymbol{\rho} : r = \max_{1\leq m\leq M} r_m < 1\}$.

In this section we show that as long as the arrival rates belong to $\mathcal{R}$, policy Π stabilizes the network in a strong sense. Specifically we show that (i) the queue length $Q_i(t)$ possesses a limiting distribution; (ii) its lth moment $EQ_i^l(t)$ as $t \to \infty$ exists provided that $ER_{ij}^{l+1} < \infty$; (iii) the queue length $Q_i(t)$ as $t \to \infty$ has an exponential tail (i.e., large backlogs are very unlikely), provided that the same is true for R_{ij}. We summarize our main results in the following proposition. Its proof is presented in the next subsection.

Proposition 1 (i) *Under policy* Π, *the process of queue lengths* $\{Q_i(t),\ i \in \mathcal{M}\}_{t=1}^{\infty}$ *converges in distribution to a random vector* $\{\widetilde{Q}_i,\ i \in \mathcal{M}\}$ *having a honest distribution, if and only if* $r = \max\{r_m : m \in \mathcal{M}\} < 1$.

(ii) *If* $ER_{ij}^{l+1} < \infty$ *for some* $l \geq 1$ *and all* $i, j \in \mathcal{M}$, *then*

$$\lim_{t\to\infty} EQ_i^l(t) = E\widetilde{Q}_i^l < \infty \tag{2}$$

(iii) *If for* $\vartheta > 0$ *the moment generating function of* R_{ij} *exists, that is,* $E\exp(\vartheta R_{ij}) < \infty$ *for all* $i, j \in \mathcal{M}$, *then there exists* $\vartheta' > 0$ *such that*

$$E\exp(\vartheta' \widetilde{Q}_i) < \infty \tag{3}$$

for every $i \in \mathcal{M}$. ■

As a direct consequence of Proposition 1(ii)-(iii), we have the following corollary concerning the tail of the queue length.

Corollary 1 (i) *If* $ER_{ij}^{l+1} < \infty$ *for all* $i, j \in \mathcal{M}$, *then the tail of the queue length distribution decays polynomially fast, that is, for some constant* $C > 0$

$$\Pr\{\widetilde{Q}_i > k\} \leq Ck^{-l}\ . \tag{4}$$

(ii) *Under the hypothesis of Proposition 1(iii), the queue length* $\widetilde{Q}_i$ *has an exponential tail, that is, there exists a constant* $C > 0$ *and* $\vartheta > 0$ *such that for all* $k \geq 0$

$$\Pr\{\widetilde{Q}_i > k\} \leq Ce^{-\vartheta k}\ . \tag{5}$$

Proof. It suffices to apply Markov inequality. For example, for part (ii) we have $\Pr\{\widetilde{Q}_i > k\} = \Pr\{e^{\vartheta\widetilde{Q}_i} > e^{\vartheta k}\} \leq e^{-\vartheta k} E e^{\vartheta \widetilde{Q}_i}$ provided (3) holds. ■

2.2 Sketch of Proofs

In this conference version of the paper, we provide only sketch of proof for the imbedded process $Q_i(\tau_n)$. Extension to continuous-time process $Q_i(t)$ can be found in our journal version [8].

The proof of Proposition 1 is based on the analysis of the so called *node degree* at time t denoted as $N_i(t)$. It is defined as the total number of packets on the ring at time t that have to cross node i in order to reach their destination. Let $N(t) := \max\{N_i(t) : i \in \mathcal{M},\ 1\}$. It was shown in [6] that $T_k := \tau_{k+1} - \tau_k = N(\tau_k)$. Now we establish an important asymptotic property of $N(\tau_2)$ that is used in the proof of Proposition 1.

Lemma 1 *If for some* $l \geq 1$ *we have* $ER_{ij}^l < \infty$ *for all* $i, j \in \mathcal{M}$, *then*

$$\lim_{n\to\infty} E\left(\left(\frac{N(\tau_2)}{T_1}\right)^l \middle| T_1 = n\right) = \max\left\{r_m^l : m \in \mathcal{M}\right\}.$$

Proof. According to the policy, the queue size at node i at time τ_2 consists of all the external packets that arrive in the interval $(1, \tau_2]$ to node i. From the definition of the degree of a node it follows that if $T_1 = n$, then $N_m(\tau_2) = \sum_{t=1}^{n} \sum_{i \in \mathcal{M}} \sum_{j=m\oplus 1}^{i} R_{ij}(t)$. Using the strong law of large numbers we conclude that

$$\lim_{n\to\infty} \frac{N_m(\tau_2)}{n} = \sum_{i\in\mathcal{M}} \sum_{j=m\oplus 1}^{i} r_{ij} = \sum_{i\in\mathcal{M}} \alpha_{im}.$$

Now let $F(\cdot)$ be a nondecreasing continuous function. In view of the above, we have for almost all sample paths,

$$\begin{aligned} \lim_{n\to\infty} F(N(\tau_2)/n) &= \lim_{n\to\infty} F\left(\max\{N_m(\tau_2)/n : \ m \in \mathcal{M}\}\right) \\ &= \max\{F(r_m) : \ m \in \mathcal{M}\}. \end{aligned} \tag{6}$$

The lemma will follow from (6) with $F(x) = x^l$, $l \geq 1$, if we show that the sequence $\{(N(\tau_2)/n)^l\}$ is uniformly integrable (u.i.). This can be accomplished by using the "mean inequality" and standard properties of uniform integrability (cf. [3]). The details can be found in [8]. ∎

A corresponding property established in Lemma 1 for the exponential tail is presented below. It is used to prove part (iii) of Proposition 1.

Lemma 2 *If* $r < 1$ *and* $E\exp(\vartheta R_{ij}) < \infty$ *for some* $\vartheta > 0$ *and all* $i, j \in \mathcal{M}$, *then there exists* $\vartheta' > 0$ *such that*

$$\lim_{n\to\infty} E\left(\left(\frac{\exp(\vartheta' N(\tau_2))}{\exp(\vartheta' T_1)} \right) | T_1 = n \right) = 0. \tag{7}$$

Proof. Observe first that if X_i, $i = 1, \ldots, K$ are random variables such that $E(\exp(\vartheta X_i)) < \infty$, $i = 1, \ldots, K$ for some ϑ, then $E\exp(\vartheta_1 \sum_{i=1}^{K} X_i) < \infty$, where $\vartheta_1 = \vartheta/K$. This is a consequence of the convexity of the exponential function (cf. [8]). Applying the previous observation to the random variables $\tilde{R}_m(t) := \sum_{i\in\mathcal{M}} \sum_{j=m\oplus 1}^{i} R_{ij}(t)$, $m \in \mathcal{M}$, we see that there is a $\vartheta_2 > 0$ such that $E\exp(\vartheta_2 \tilde{R}_m(t)) < \infty$ for $m \in \mathcal{M}$.

Consider now the function $\Phi_m(\vartheta) = E\exp(\vartheta \tilde{R}_m(t) - \vartheta)$, $0 \leq \vartheta \leq \vartheta_2$. From the previous discussion we see that this function is well defined, continuous and differentiable in $[0, \vartheta_2]$. Since $\Phi_m(0) = 1$, $\Phi'_m(0) = E\tilde{R}_m(t) - 1 = r_m - 1 \leq r - 1 < 0$, and $M < \infty$, it follows that there is a $\vartheta' > 0$ and a $\epsilon > 0$, such that for $m \in \mathcal{M}$, $\Phi_m(\vartheta') < 1 - \epsilon$, or equivalently, $\exp(-\vartheta')E\exp(\vartheta' \tilde{R}_m(t)) < (1 - \epsilon)$. Since the random variable $R_m(t)$, $t = 1, \ldots$ are i.i.d, we conclude that

$$\frac{E\exp(\vartheta' N_m(\tau_2))}{\exp(\vartheta' n)} = \left(\frac{E\exp(\vartheta' R_m(t))}{\exp(\vartheta')} \right)^n \leq (1 - \epsilon)^n.$$

But, $E\exp(\vartheta' N(\tau_2)) \leq \sum_{m\in\mathcal{M}} E\exp(\vartheta' N_m(\tau_2))$, hence (7) follows. ∎

From Lemmas 1 and 2 we easily conclude our next result.

Corollary 2 (i) *If* $r < 1$ *and* $ER_{ij}^l < \infty$ *for all* $i, j \in \mathcal{M}$ *and some* $l \geq 1$, *then there exist* $\delta > 0$ *and* $B > 0$ *such that*

$$E(N^l(\tau_2) | N(\tau_1) = n) \leq (1 - \delta) n^l \qquad n \geq B\,.$$

(ii) *If* $r < 1$ *and* $E\exp(\vartheta R_{ij}) < \infty$ *for some* $\vartheta > 0$ *and all* $i, j \in \mathcal{M}$, *then there exists* $\vartheta' > 0$ *such that*

$$E\left(\exp(\vartheta' N(\tau_2)) | N(\tau_1) = n\right) \leq (1 - \delta)\exp(\vartheta' n) \qquad n \geq B\,.$$

Proof. Since $r_m \leq r < 1$, we have that $\max\{r_m^l : \ m \in \mathcal{M}\} \leq r < 1 - \delta$, $\delta > 0$. Using this observation, part (i) of the Corollary 2 follows directly from from Lemma 1. Part (ii) follows directly from Lemma 2. ∎

To proceed, we need the following theorem, due to Tweedie, [10, Theorem 3] , which we present in a form appropriate for the problem under consideration.

Theorem 1 (Tweedie.) *Suppose that* $\{X_n\}_{n=1}^{\infty}$ *is an aperiodic and irreducible Markov chain with countable state space* S. *Let* $f(x)$ *be a nonnegative real function on the state space. If* A *is a finite set such that* $f(x) \geq \epsilon > 0$, *and* $E(f(X_2)|X_1 = x) < \infty$ *for* $x \in A$, *and for some* $\delta > 0$,

$$E(f(X_2)|X_1 = x) < (1-\delta)f(x), \quad x \in A^c,$$

then the Markov chain is ergodic and $Ef(\hat{X}) < \infty$, *where* $\hat{X}$ *has the steady state distribution of the Markov chain* $\{X_n\}_{n=1}^{\infty}$. ■

Let $Q_{ij}(t)$ be the number of packets at node i with destination node j, including both external packets and packets received from the incoming link to node i. Clearly, $Q_i(t) = \sum_{j \in \mathcal{M}} Q_{ij}(t)$. Let also $\mathbf{Q}(t) := \{Q_{ij}(t) : i, j \in \mathcal{M}\}$. From the operation of the policy and assumption (A1) we conclude that the process $\{\mathbf{Q}(\tau_n)\}_{n=1}^{\infty}$ is an imbedded Markov chain. Using Theorem 1 we shall prove that this imbedded Markov chain is ergodic if $r < 1$. Moreover, using a regenerative structure of the queueing process, we can extend this assessment to all t, hence proving our main result Proposition 1 (cf. [8]).

Let us only consider the imbedded Markov chain $\{\mathbf{Q}(\tau_n)\}_{n=1}^{\infty}$. We prove that under the condition of Proposition 1 this process converges weakly to a honest random vector $\hat{\mathbf{Q}}$. Since by assumption $\Pr(R_{ij}(t) = 0, \ i,j \in \mathcal{M}) > 0$, it can be seen that the imbedded Markov chain has only one irreducibility set and if restricted to this set, the chain is aperiodic.

Theorem 2 (i) *If* $r < 1$ *then the Markov chain* $\{\mathbf{Q}(\tau_n)\}_{n=1}^{\infty}$ *is ergodic.* (ii) *If, in addition, for* $l \geq 2$ *we have* $ER_{ij}^l < \infty$ *for all* $i, j, \in \mathcal{M}$, *then* $\lim_{n\to\infty} EQ_i^l(\tau_n) < \infty$. (iii) *If for some* $\vartheta > 0$ *we have* $E\exp(\vartheta R_{ij}) < \infty$ *for all* $i, j \in \mathcal{M}$, *then there exists* $\vartheta' > 0$ *such that* $\lim_{n\to\infty} E\exp(\vartheta' Q_i(\tau_n)) < \infty$.

Proof. Define first $f_l(\mathbf{Q}(\tau_n)) = N^l(\tau_n), \quad l \geq 1$. Clearly, the set $A := \{\mathbf{Q} : f_l(\mathbf{Q}) < B\}$ is finite. Also, if $\mathbf{Q}(\tau_1) \in A$, then since $T_1 = N(\tau_1) < B^{(1/l)}$, using arguments similar to those used in the proof of Lemma 1 it can be easily seen that $E\left(N^l(\tau_2)|\mathbf{Q}(\tau_1) = \mathbf{Q} \in A\right) < \infty$, provided that $ER_{ij}^l < \infty$. From the above discussion, Corollary 2 and Theorem 1 we conclude that $\{\mathbf{Q}(\tau_n)\}_{n=1}^{\infty}$ is ergodic, and provided that $ER_{ij}^l < \infty, \ i,j \in \mathcal{M}$ for some $l \geq 1$, we have $E\hat{N}^l < \infty$, where $\hat{N}^l = f_l(\hat{\mathbf{Q}})$, and $\hat{\mathbf{Q}}$ has the steady state distribution of $\{\mathbf{Q}(\tau_n)\}_{n=1}^{\infty}$. For part (iii), the proof is along the same lines with $f(\mathbf{Q}(\tau_n)) = \exp(\vartheta N(\tau_n))$. ■

3 Stability of Ring with Fixed Quota

In this section we consider the ring with finite quota under the same mathematical model as in the study of the adaptive policy, however, we strengthen the statistical assumptions by postulating that the packet arrival processes at the nodes, as well as the processes of packet destinations are independent. Specifically, let $R_i(t)$ be the number of packets that arrived at station i in slot t. The nth packet arriving at node i has destination $d_i(n)$. The processes $\{R_i(t)\}_{t=1}^{\infty}$, $\{d_i(n)\}_{n=1}^{\infty}$ consist of i.i.d. random variables and are independent of each other for all $i \in \mathcal{M}$. We set $\lambda_i = ER_i(1)$ and $q_{ij} = \Pr(d_i(1) = j)$.

The system operates again in cycles, but in contrast to the adaptive policy, there is now an upper bound, $\hat{\nu}_i$, on the number of packets that node i can transmit in a cycle. Specifically, the number of packets node i transmits in the kth cycle is $\nu_i(k) = \min\{\hat{\nu}_i, \ Q_i(\tau_k)\}$, where τ_k is the time the kth cycle begins and $Q_i(t)$ is the number of packets queued at node i at time t. Let $\boldsymbol{\nu}^{\mathcal{M}}(k) := \{\nu_i(k) : \ i \in \mathcal{M}\ \}$ and denote by $T(\boldsymbol{\nu}^{\mathcal{M}}(k))$ the cycle length when node i transmits $\nu_i(k)$ packets in that cycle. For technical reasons that will become apparent in the derivation of the stability region, we will consider a system π that contains, in addition to the regular nodes in the set $\mathcal{M}$, a set $\mathcal{U}$ of nodes with the following properties. 1) There is no packet arrival to a node $i \in \mathcal{U}$. 2) A node $i \in \mathcal{U}$ always transmits $\hat{\nu}_i$ "dummy" packets in a cycle. While the nodes in $\mathcal{U}$ affect the duration of cycles, by definition they do not have queues and we interested only in the stability of the process of queue lengths of the nodes in $\mathcal{M}$. We will refer to the nodes in $\mathcal{U}$ as the "persistent" nodes and to the nodes in $\mathcal{M}$ as the "nonpersistent" nodes.

Consider now a modified system, $\pi^{\mathcal{P}}$, which contains the same nodes as system π and operates identically with system π except that for a subset $\mathcal{P} \subseteq \mathcal{M}$, node $i \in \mathcal{P}$ always transmits $\hat{\nu}_i$ packets in a cycle, by transmitting if necessary $\hat{\nu}_i - Q_i(\tau_k)$ "dummy" packets during cycle k. Therefore, the nodes in $\mathcal{M} - \mathcal{P}$ under system $\pi^{\mathcal{P}}$ operate as a regular system π that contains $\mathcal{P} \cup \mathcal{U}$ persistent nodes. Note, however, that queues are still formed at node $i \in \mathcal{P}$ under system $\pi^{\mathcal{P}}$.

Using the same method as in the proof of Theorem 4 in [7] we have the following result for the system under consideration.

Proposition 2 *Let $\mathcal{P}_1 \subseteq \mathcal{P}_2 \subseteq \mathcal{M}$ and assume that systems $\pi^{\mathcal{P}_1}$, $\pi^{\mathcal{P}_2}$ start with the same initial conditions, i.e., $Q_i^{\mathcal{P}_1}(1) = Q_i^{\mathcal{P}_2}(1)$. Then, $Q_i^{\mathcal{P}_1}(\tau_k) \leq_{st} Q_i^{\mathcal{P}_2}(\tau_k)$, for all k and $i \in \mathcal{M}$.* ■

Using Proposition 2 we can establish a recursive way of providing the sufficient conditions for the stability of system π, along the lines presented in [9]. We will only outline the method here. We start by considering the sufficient conditions for the stability of a system with one nonpersistent node and an arbitrary but fixed set $\mathcal{U}_1$ of persistent nodes. Next, provided that we establish the sufficient conditions for the stability of a system with $M - 1$ nonpersistent nodes and a set $\mathcal{U}_{M-1}$ of persistent nodes, we will find the sufficient conditions for the stability of the system with M nonpersistent nodes and $\mathcal{U}_M$ persistent nodes. As will be seen, in general the stability conditions depend on the arrival distributions, on the quota sizes and the packet destination probabilities. Of course, the stability of the original system that contains only M nonpersistent nodes can be obtained by setting $\mathcal{U}_M = \emptyset$

Consider first the system with 1 nonpersistent node, say node i, and a set $\mathcal{U}_1$ of persistent nodes. In this case the process $\{Q_i(\tau_k)\}_{k=1}^{\infty}$ is a Markov chain and standard arguments based on Foster's criteria show that the persistent node is stable when

$$0 \leq \lambda_i < \frac{\hat{\nu}_i}{ET^{\{i\},\mathcal{U}_1}} \tag{8}$$

where $ET^{\{i\},\mathcal{U}_1}$ is the average length of a cycle in such a system.

Assume now that we have established the stability region for a system with a set $\mathcal{S}_{M-1}$ of nonpersistent nodes such that $|\mathcal{S}_{M-1}| = M - 1$, and a set $\mathcal{U}_{M-1}$ of persistent nodes. Let this region be of the form $\{\lambda_j\}_{j \in \mathcal{S}_{M-1}} \in \mathcal{R}^{\mathcal{S}_{M-1},\mathcal{U}_{M-1}}$. We will find next the corresponding conditions for the system π with a set $\mathcal{S}_M$ of nonpersistent nodes and a set $\mathcal{U}_M$ of persistent nodes. Consider the modified system $\pi^{\{i\}}$ that differs from π only in the fact that node $i \in \mathcal{S}_M$ always transmits $\hat{\nu}_i$ packets in a cycle by transmitting, if necessary, dummy packets. From Proposition 2 we have that the queue sizes of the nodes in the set $\mathcal{S}_M$ at the beginning of the kth cycle in system π are stochastically smaller than the corresponding queue sizes of the same set of nodes at the beginning of the kth cycle in system $\pi^{\{i\}}$. Therefore, the sufficient condition for the stability of the set of nodes $\mathcal{S}_M$ in system $\pi^{\{i\}}$ is also a sufficient condition for the stability of the same set of nodes in system π. To establish the stability of the M-dimensional system $\pi^{\{i\}}$ it is sufficient to establish the stability of each of the 1-dimensional queues $\{Q_j^{\pi^{\{i\}}}(k)\}_{k=1}^{\infty}$, for all $j \in \mathcal{S}_M$ (cf. [7], [9]). Now, the set of $\mathcal{S}_M^i = \mathcal{S}_M - \{i\}$ nodes in system $\pi^{\{i\}}$ operates as a system of $M - 1$ nonpersistent nodes and a set $\mathcal{U}_M^i = \mathcal{U}_M \cup \{i\}$ of persistent nodes. Therefore, the stability condition of $\{Q_j^{\pi^{\{i\}}}(k)\}_{k=1}^{\infty}$ for $j \in \mathcal{M}^i$ is known by the inductive construction and is of the form $\{ \{\lambda_j\}_{j \in \mathcal{S}_M^i} \in \mathcal{R}^{\mathcal{S}_M^i,\mathcal{U}_M^i}\}$. Next we turn our attention to the queue at node i, $\{Q_i^{\pi^{\{i\}}}(k)\}_{k=1}^{\infty}$. Provided that $\{\lambda_j\}_{j \in \mathcal{S}_M^i} \in \mathcal{R}^{\mathcal{S}_M^i,\mathcal{U}_M^i}$, it can be established that the cycle lengths have a stationary distribution and that under appropriate initial conditions, the sequence of the cycle lengths is strictly stationary. Let $\overline{T}^{\mathcal{S}_M^i,\mathcal{U}_M^i}$ be the steady-state cycle length in this case.

We can now view node i as a single queue with stationary inputs, and following the techniques of Loynes, we can establish that $\{Q_i^{\pi^{\{i\}}}(k)\}_{k=1}^{\infty}$ is stable if $\lambda_i < \hat{\nu}_i / E\overline{T}^{\mathcal{S}^i_M, \mathcal{U}^i_M}$. Therefore, system $\pi^{\{i\}}$ and consequently system π, is stable for $\lambda_i < \hat{\nu}_i / E\overline{T}^{\mathcal{S}^i_M, \mathcal{U}^i_M}$ and $\{\lambda_j\}_{j \in \mathcal{S}^i_M} \in \mathcal{R}^{\mathcal{S}^i_M, \mathcal{U}^i_M}$. This leads to the following characterization of the stability region.

Proposition 3 *System π is stable in the region $\mathcal{R}^{\mathcal{S}_M, \mathcal{U}_M}$ given by*

$$\mathcal{R}^{\mathcal{S}_M, \mathcal{U}_M} = \bigcup\nolimits_{i \in \mathcal{M}} \left\{ \{\lambda_j\}_{j \in \mathcal{S}_M} : 0 \leq \lambda_i < \frac{\hat{\nu}_i}{E\overline{T}^{\mathcal{S}^i_M, \mathcal{U}^i_M}} , \; \{\lambda_j\}_{j \in \mathcal{S}^i_M} \in \mathcal{R}^{\mathcal{S}^i_M, \mathcal{U}^i_M} \right\}. \tag{9}$$

When $\mathcal{U} = \emptyset$ one obtains the stability region for the original system. ■

Following a methodology similar to [9] it can be shown that except the boundary points, these conditions are also necessary for stability.

Remark. Using equation (9) we can describe the stability condition in a more explicit form as follows. Let $\theta(k) : \mathcal{M} \to \mathcal{S}_M$ be a permutation of the set $\mathcal{S}_M$. Define $\mathcal{S}_M^\theta(1) = \emptyset$ and for $k \geq 2$, $\mathcal{S}_M^\theta(k) = \cup_{n=1}^{k-1} \{\theta(n)\}$. Let also, $\mathcal{U}_M^\theta(k) = (\mathcal{S}_M - \mathcal{S}_M^\theta(k)) \cup \mathcal{U}_M$. Then,

$$\mathcal{R}^{\mathcal{S}_M, \mathcal{U}_M} = \bigcup_{\theta} \left\{ \{\lambda_j\}_{j \in \mathcal{S}_M} : 0 \leq \lambda_{\theta(k)} < \frac{\hat{\nu}_{\theta(k)}}{E\overline{T}^{\mathcal{S}_M^\theta(k), \mathcal{U}_M^\theta(k)}} , \; k \in \mathcal{M} \right\}, \tag{10}$$

where the union is over all permutations. □

The computation of the stability region as described in (9) or (10) becomes quickly intractable. This is due to the fact that in general, the stability region for a system with M nodes depends on the steady state probabilities of stable lower dimensional systems which in turn depend not only on the arrival rates but also on the distribution of the arrival processes to the nodes. The following example illustrates this point.

Example 1. Consider the ring with quotas $\hat{\nu}_1 = 2$, $\hat{\nu}_2 = 1$ and $\mathcal{U}_2 = \emptyset$. Let us also assume the simplest destination probabilities, that is, $q_{12} = q_{21} = 1$. In this case the stability region can be expressed as follows.

$$\mathcal{R}^{\{1,2\}, \emptyset} = \left\{ \lambda_1 < \frac{2}{E\overline{T}^{\{2\},\{1\}}}, \; \lambda_2 < \frac{1}{E\overline{T}^{\emptyset,\{1,2\}}} \right\} \cup \left\{ \lambda_1 < \frac{2}{E\overline{T}^{\emptyset,\{1,2\}}}, \; \lambda_2 < \frac{1}{E\overline{T}^{\{1\},\{2\}}} \right\} \tag{11}$$

The computation of the first set on the right hand side in (11) is straightforward. Indeed, observe that by the choice of the destination probabilities, a node never has to retransmit packets originated by the other node. The interaction between the two queues is due only to the fact that one node may have to wait until the other one completes transmission of its quota. Therefore, if node 1 is persistent, the cycle length will be 2 irrespective of the status of node 2, that is, $E\overline{T}^{\emptyset,\{1,2\}} = E\overline{T}^{\{2\},\{1\}} = 2$, and the first region becomes $\{0 \leq \lambda_1 < 1, \; 0 \leq \lambda_2 < .5\}$.

We consider now the second set in (11). The quantity that needs to be determined in this case is the expected cycle length in steady state, when node 2 is persistent and node 1 is nonpersistent, that is, $E\overline{T}^{\{1\},\{2\}}$. Assume therefore that node 2 is persistent and let $Q_k = Q_1(\tau_k)$ be the queue size at node 1 at the beginning of the kth cycle. Let $R(z)$ be the z-transform of $R_1(1)$, the number of arrivals to node 1 in the first slot. (Recall that we assume that $\{R_1(k)\}_{k=1}^{\infty}$ are i.i.d.). Let $l = \tau_k$, be the time when the kth cycle starts. Then, it is easy to see that

$$Q_{k+1} = (Q_k - 2)^+ + R_l + R_{l+1} \mathbf{1}_{\{Q_k \geq 2\}}. \tag{12}$$

Based on (12) the z-transform of the steady-state distribution of Q_k can be derived and based on this we conclude that

$$E\overline{T}^{\{1\},\{2\}} = \frac{4 z_a}{2 z_a + (1 - \lambda_1) z_a - (1 - \lambda_1)}$$

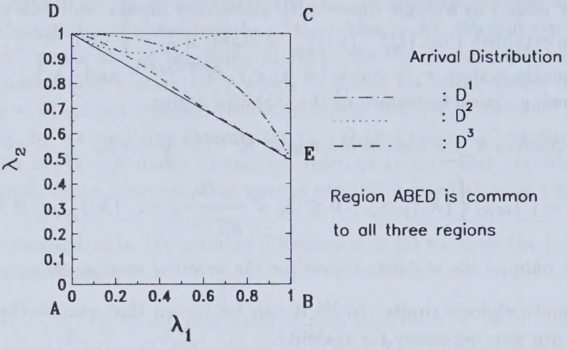


Figure 1: Stability region for the ring of Example 1.

where z_a is the unique root in $[-1, 0)$ of the equation $R(z) = -z$ Therefore, the second set in (11) is equal to

$$\left\{0 \le \lambda_1 < 1,\ 0 \le \lambda_2 < \frac{2z_a + (1-\lambda_1)z_a - (1-\lambda_1)}{4z_a}\right\}$$

The root z_a depends on the distribution of the arrival process to node 1 and as a result the same is true for the stability region. To demonstrate this strong dependence, we plotted in Figure 1 the stability regions regions for the following arrival distributions to node 1

1. $\mathbf{D}^1 = \{\Pr(R_1(1) = 0), \Pr(R_1(1) = 1), \Pr(R_1(1) = 2)\} = \{\alpha,\ \ 2\alpha(1-\alpha),\ \ (1-\alpha)^2\}$, $0 \le \alpha \le 1$. This is binomial with parameters $\alpha, 2$
2. $\mathbf{D}^2 = \{1 - \alpha e^{-1/(1-\alpha)} - .5\alpha^2,\ \ \alpha e^{-1/(1-\alpha)},\ \ .5\alpha^2\},\ \ 0 \le \alpha \le 1$
3. $\mathbf{D}^3 = \{1 - \alpha^2 - \alpha e^{-1/(1-\alpha)},\ \ \alpha^2,\ \ \alpha e^{-1/(1-\alpha)}\},\ 0 \le \alpha \le 1$

In Figure 1 the region $ABED$ is common for the three arrival distributions. However, the rest of the region depends strongly on the arrival distribution. Note that the stability region of the adaptive policy for this system is the region $ABCD$. As we will see in the following, the region $ABED$ is a subset of the stability region for *any* arrival distribution. □

Motivated by the strong dependence of the stability region on the arrival rate distribution, we define the following stability regions of arrival rates.

Effective Stability Region (ESR): The set of arrival rates $(\lambda_1, ..., \lambda_M)$ with the property that the system is stable under *any* arrival distributions as long as they have the corresponding rates $(\lambda_1, ..., \lambda_M)$.

Absolute Stability Region (ASR): The set of arrival rates $(\lambda_1, ..., \lambda_M)$ with the property that there is *at least one* set of node arrival distributions with corresponding rates $(\lambda_1, ..., \lambda_M)$ such that the system is stable.

Clearly, ESR⊂ASR. The ESR is useful in situations where the arrival distributions are not known *apriori*, a common situation in many practical systems. Besides the theoretical interest of the ASR as the region outside which the system cannot be stable under any arrival distribution, it might also have practical implications when the input traffic can be controlled before entering the network.

Based on (9) and (10), we will now develop lower and upper bounds for the ESR and ASR respectively, that depend only on the conditional cycle lengths, $ET(\boldsymbol{\nu})$, which can be computed without knowing the steady state probabilities. The bounds are derived by associating the stability of the system with the solutions of linear programming optimization problems whose constraints are derived from equilibrium of flow arguments. Let $\mathcal{S} \neq \emptyset$ and consider the stability region $\mathcal{R}^{\mathcal{S},\mathcal{U}}$. When $\{\lambda_j\}_{j\in\mathcal{S}_M} \in \mathcal{R}^{\mathcal{S},\mathcal{U}}$, the nodes in the set $\mathcal{S}$ constitute a stable system. Let $\pi(\mathbf{n})$, $\mathbf{n} = \{n_j,\ j \in \mathcal{S}\}$ be the steady state probability of the process of node queue lengths at the beginning of a cycle and for $l \in \mathcal{S}$ define the one dimensional distribution $\pi_l(n) = \sum_{\mathbf{n},\, n_l=n} \pi(\mathbf{n})$. Standard arguments based on the regenerative theorem can be used to show that the following flow equations are satisfied for the system consisting of the nodes in $\mathcal{S}$.

$$\lambda_l E\overline{T}^{\mathcal{S},\mathcal{U}} = \sum_{n=1}^{\infty} f_l(n)\pi_l(n), \qquad l \in \mathcal{S}, \tag{13}$$

where $f_l(n)$ is the number of packets transmitted by node l when the number of packets at that node at the beginning of a cycle is n. Let now $ET(\mathbf{m},\hat{\boldsymbol{\nu}}\}$, $\mathbf{m} = \{m_j :\ j \in \mathcal{S}\}$, $\hat{\boldsymbol{\nu}} = \{\hat{\nu}_j :\ j \in \mathcal{U}\}$ be the average cycle length when node $j \in \mathcal{S}$ transmits m_j packets in a cycle, and node $j \in \mathcal{U}$ transmits $\hat{\nu}_j$ packets. Then the average steady state cycle length satisfies, $E\overline{T}^{\mathcal{S},\mathcal{U}} = \sum_{\mathbf{n}} ET(\mathbf{f}(\mathbf{n}),\hat{\boldsymbol{\nu}})\pi(\mathbf{n})$, where $\mathbf{f}(\mathbf{n}) = \{f_j(n_j),\ j \in \mathcal{S}\}$.

Let us now define for $\mathbf{n} = \{n_j,\ j \in \mathcal{S}\}$, $0 \le n_j \le \hat{\nu}_j$

$$x(\mathbf{n}) = \begin{cases} \pi(\mathbf{n}) & \text{if } \ 0 \le n_j \le \hat{\nu}_j - 1,\ j \in \mathcal{S} \\ \sum_{n_j \ge \hat{\nu}_j,\, j\in\mathcal{G}} \pi(\mathbf{n}) & \text{if } \ n_j = \hat{\nu}_j,\ j \in \mathcal{G} \subseteq \mathcal{S},\ 0 \le n_j \le \hat{\nu}_j - 1,\ j \in \mathcal{S} - \mathcal{G}. \end{cases}$$

In terms of these variables, we can rewrite the above equations as follows

$$\left.\begin{array}{rcl} \lambda_l \sum_{\mathbf{n}} ET(\mathbf{f}(\mathbf{n}),\hat{\boldsymbol{\nu}})x(\mathbf{n}) & = & \sum_{n=1}^{\hat{\nu}_l} n \sum_{\mathbf{n},\, n_l=n} x(\mathbf{n}), \quad l \in \mathcal{S} \\ \sum_{\mathbf{n}} x(\mathbf{n}) & = & 1 \\ x(\mathbf{n}) & \ge & 0\ . \end{array}\right\} \tag{14}$$

Clearly, we also have $E\overline{T}^{\mathcal{S},\mathcal{U}} = \sum_{\mathbf{n}} ET(\mathbf{f}(\mathbf{n}),\hat{\boldsymbol{\nu}})x(\mathbf{n})$. From the above discussion we see that with each pair $\mathcal{S},\mathcal{U}$ we can associate a polytope $\wp^{\mathcal{S},\mathcal{U}}$ defined by the constraints in (14). Let us define $T_{\max}^{\emptyset,\mathcal{U}} = ET(\hat{\boldsymbol{\nu}})$ and for $\mathcal{S} \neq \emptyset$, define $T_{\max}^{\mathcal{S},\mathcal{U}}$ as the solution, if it exists, of the following linear programming optimization problem

$$T_{\max}^{\mathcal{S},\mathcal{U}} = \arg \max_{x(\mathbf{n})\in\wp^{\mathcal{S},\mathcal{U}}} \left\{ \sum_{\mathbf{n}} ET(\mathbf{f}(\mathbf{n}),\hat{\boldsymbol{\nu}})x(\mathbf{n}) \right\} \tag{15}$$

Notice that the solution of this optimization problem requires only the knowledge of the average values of the conditional cycle lengths. Using the notation of equation (10) and replacing for notational simplicity $\mathcal{S}_M$, $\mathcal{U}_M$ with $\mathcal{S}$, $\mathcal{U}$ respectively, define

$$\begin{aligned} \mathcal{R}_\theta^{\mathcal{S}^\theta,\mathcal{U}^\theta} &= \left\{ \{\lambda_j\}_{j\in\mathcal{S}^\theta} :\ 0 \le\ \lambda_{\theta(k)} < \frac{\hat{\nu}_{\theta(k)}}{E\overline{T}^{\mathcal{S}^\theta(k),\mathcal{U}^\theta(k)}},\ k \in \mathcal{M} \right\}, \\ \tilde{\mathcal{R}}_\theta^{\mathcal{S}^\theta,\mathcal{U}^\theta} &= \left\{ \{\lambda_j\}_{j\in\mathcal{S}^\theta} :\ 0 \le\ \lambda_{\theta(k)} < \frac{\hat{\nu}_{\theta(k)}}{T_{\max}^{\mathcal{S}^\theta(k),\mathcal{U}^\theta(k)}},\ k \in \mathcal{M} \right\}. \end{aligned}$$

It can be established that for any permutation $\theta : \mathcal{M} \to \mathcal{S}, \tilde{\mathcal{R}}_\theta^{\mathcal{S}^\theta,\mathcal{U}^\theta} \subseteq \mathcal{R}_\theta^{\mathcal{S}^\theta,\mathcal{U}^\theta}$, and therefore, $\bigcup_\theta \tilde{\mathcal{R}}^{\mathcal{S}^\theta,\mathcal{U}^\theta} \subseteq \mathcal{R}^{\mathcal{S}^\theta,\mathcal{U}^\theta}$. Note that since $\bigcup_\theta \tilde{\mathcal{R}}^{\mathcal{S}^\theta,\mathcal{U}^\theta}$ is independent of the arrival distribution, it is in fact a subset of the ESR. In a similar fashion a superset of the ASR can be derived.

Let us now consider two nodes ring, that is, $\mathcal{S} = \{1,2\}$ and $\mathcal{U} = \emptyset$,. In this case, the effective stability region can be computed explicitly in terms of the conditional lengths for arbitrary quota sizes. This is due to the following property of the conditional cycle lengths.

Lemma 3 *The functions*

$$\varphi(n) := \frac{ET(\hat{\nu}_1, n) - ET(\hat{\nu}_1, 0)}{n}, \qquad 1 \leq n \leq \hat{\nu}_2,$$

$$\psi(n) := \frac{ET(n, \hat{\nu}_2) - ET(0, \hat{\nu}_2)}{n}, \qquad 1 \leq n \leq \hat{\nu}_1,$$

are nondecreasing. ■

Based on Lemma 3 and using the bound for the ESR derived in the previous section we can determine the ESR for the ring with two nodes. (Proofs are delayed till a journal version of the paper.)

Proposition 4 *The effective stability region for the ring with finite quota is the set*

$$\begin{aligned} \mathcal{R} &= \left\{ \{\lambda_i\}_{i=1,2} : \lambda_1 < \tfrac{\hat{\nu}_1}{ET(\hat{\nu}_1,\hat{\nu}_2)}, \left(\tfrac{ET(\hat{\nu}_1,\hat{\nu}_2)}{\hat{\nu}_1} - \tfrac{\hat{\nu}_2}{\hat{\nu}_1} \right) \lambda_1 + \lambda_2 < 1, \ \lambda_i \geq 0 \right\} \\ &\cup \ \left\{ \{\lambda_i\}_{i=1,2} : \lambda_2 < \tfrac{\hat{\nu}_2}{ET(\hat{\nu}_1,\hat{\nu}_2)}, \ \lambda_1 + \left(\tfrac{ET(\hat{\nu}_1,\hat{\nu}_2)}{\hat{\nu}_2} - \tfrac{\hat{\nu}_1}{\hat{\nu}_2} \right) \lambda_2 < 1, \ \lambda_i \geq 0 \right\} \end{aligned}$$

where $ET(\hat{\nu}_1, \hat{\nu}_2)$ can be computed solely from the distribution of the arrival processes . ■

References

[1] S. Asmussen. *Applied Probability and Queues.* John Wiley and Sons, 1987.

[2] N. Bambos and A. Nguyen, Optimal message flow on ring network with spatial reuse, *Proc. 31-st Conference on Decision and Control*, pp. 2360-2363, Tucson, AZ, December 1992.

[3] Y. S. Chow and H. Teicher. *Probability Theory: Independence, Interchangeability, Martingales.* Springer Texts in Statistics. Springer Verlag, 1988.

[4] I. Cidon and Y. Ofek. MetaRing - a full-duplex ring with fairness and spatial reuse. *IEEE Trans. Commun.*, 41, 110–120, 1993.

[5] R. M. Falconer and J. L. Adams. Orwell: A protocol for an integrated service local network. *Br. Telecom Technol. J.*, 3(4):21–35, April 1985.

[6] L. Georgiadis, R. Guerin, and I. Cidon. Throughput properties of fair policies in ring networks. Technical Report RC 18293(80185), IBM, September 1992.

[7] L. Georgiadis and W. Szpankowski. Stability of token passing rings. *Queueing System*, 11, 7-33, 1992.

[8] L. Georgiadis, W. Szpankowski and L. Tassiulas, Adaptive policies for spatial reuse ring networks, INRIA TR, 1993.

[9] W. Szpankowski Stability conditions for some multiqueue distributed systems *Advances in Applied Probability* to appear, June 1994.

[10] R. L. Tweedie. The existence of moments for stationary Markov chains. *Journal of Applied Probability*, 20:191–196, 1983.

Structural Properties of the Throughput in Cyclic Networks With General Blocking And Starvation*

Rajendran Rajan and Rajeev Agrawal

Department of Electrical and Computer Engineering
University of Wisconsin–Madison
Madison, WI 53706-1691, USA
E-mail: agrawal@engr.wisc.edu

Abstract

We introduce *general starvation* and consider cyclic networks with *general blocking and starvation* (GBS). The mechanism of general blocking (with parameter b) allows the server to process upto b jobs when the buffer downstream is full, and that of general starvation (with parameter c) allows the server to perform upto c services in anticipation of jobs that are yet to arrive. Apart from increased modelling power, there is a significant theoretical motivation behind incorporating general starvation into the queueing model. General starvation turns out to be the *dual* of general blocking. We exploit this duality crucially, to extend several results that were hitherto available only for communication blocking networks to a much larger class of GBS networks. The main aim of this paper is to investigate how the throughput of a cyclic GBS network is affected by varying the total number of jobs J. First, for cyclic GBS networks where the blocking and starvation parameters at each station are equal, we show that the throughput is symmetric in J and is maximized when $J = K/2$, where K is the number of buffer spaces available. Next, for two-station GBS cyclic networks with $b_1 - c_1 = b_2 - c_2 = \Delta$, we show that the throughput is symmetric about $K/2 + \Delta$. This result, even for networks without general starvation ($c_1 = c_2 = 0$) makes use of general starvation in an essential way. In the course of proving these symmetry and throughput maximization results, we establish several intermediate structural properties of the throughput, using duality, reversibility, and concavity, which are of independent interest.

1. The General Blocking and Starvation (GBS) Model

We consider cyclic networks of stations with *general blocking and starvation* (GBS). As shown in Figure 1, each station of a cyclic GBS network comprises of a buffer for jobs, a buffer for *tokens*, and a server which can be thought of as a token generator. The jobs are token consumers, in the sense that at each station a job needs a token to move on to the next station. There are three parameters associated with any station $p : (k_p, b_p, c_p)$, where k_p is the size of the job buffer, b_p is the *blocking* parameter, and c_p is the *starvation* parameter. The meaning of these parameters will be made clear shortly. The state of this system at any time $t \geq 0$ is specified by $(\mathbf{J}(t), \mathbf{L}(t)) = ((J_1(t), \ldots, J_m(t)), (L_1(t), \ldots, L_m(t)))$, where $J_p(t)$ and $L_p(t)$ represent the number of jobs and tokens respectively at station $p = 1, \ldots, m$ at time t. Let the initial state be given by $(\mathbf{J}(0), \mathbf{L}(0)) = (\mathbf{j}^0, \mathbf{l}^0)$. It is

---

*Research supported in part by NSF Grant No. ECS-8919818.

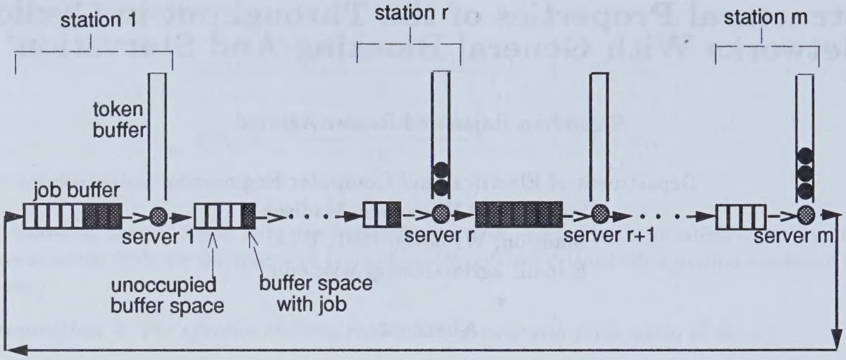


Figure 1: A Cyclic Network with General Blocking and Starvation.

natural to impose the following conditions on the parameters:

$$0 \leq J_p(t) \leq k_p, \qquad 0 < \sum_{p=1}^{m} J_p(t) = J < \sum_{p=1}^{m} k_p = K, \qquad L_p(t) \geq 0, \tag{1.1}$$

$$J_p(t) L_p(t) (k_{p\oplus 1} - J_{p\oplus 1}(t)) = 0. \tag{1.2}$$

The condition (1.2) is derived from the rule that if there is a job and a token at station p, and an empty buffer space (*hole*) at the station $p \oplus 1$ downstream then the job uses up the token and moves instantaneously to the station downstream. We also impose the following two constraints on L_p:

$$\begin{aligned} L_p(t) &\leq b_p + k_{p\oplus 1} - J_{p\oplus 1}(t), & (1.3) \\ L_p(t) &\leq c_p + J_p(t). & (1.4) \end{aligned}$$

The server is said to be *blocked* if (1.3) holds with equality, and is said to be *starved* if (1.4) holds with equality. The server is *disabled*, i.e., stops generating tokens if it is either blocked or starved. Otherwise the server is *enabled.* Note that if the job buffer of the next station downstream is full, i.e., $J_{p\oplus 1}(t) = k_{p\oplus 1}$, then the server can generate at most b_p more tokens. Thus b_p characterizes blocking at station p. On the other hand, if there are no jobs at station p, i.e., $J_p(t) = 0$, then the server can generate at most c_p more tokens. Thus c_p generalizes the notion of starvation by allowing services in anticipation. It is this starvation parameter that distinguishes our model from the other models considered in the literature so far.

When a token is generated at station p, the state of the system changes as follows. First a token is placed in the token buffer at station p. If there is a job at station p, then this job is moved as far downstream as possible by consuming the corresponding tokens. Finally, if there is a hole at station p (as a result of a job moving downstream), then this hole is moved as far upstream as possible, again by consuming the appropriate tokens. Moving a hole from station p to $p \ominus 1$ is equivalent to moving a job from station $p \ominus 1$ to p. It is easy to check that the resulting state satisfies (1.1), (1.2), (1.3), and (1.4).

The cyclic GBS network is driven by the sequence of service times $\omega = \{\omega_p^i \ : \ 1 \leq p \leq m, \ i \geq 1\}$, where ω_p^i is the service time of the i-th job to be processed at station

p. The service time sequence ω may be random, in which case the resulting state process is a generalized semi-Markov process (GSMP) (see [15], [11], [10] for details). The cyclic GBS network $S = S((\mathbf{k},\mathbf{b},\mathbf{c}),(\mathbf{j}^0,\mathbf{l}^0),\omega)$ is completely defined by the parameters $((\mathbf{k},\mathbf{b},\mathbf{c}),(\mathbf{j}^0,\mathbf{l}^0))$ and the process ω. We shall refer to $(\mathbf{k},\mathbf{b},\mathbf{c})$ as the *GBS parameters* of S, and to $(\mathbf{j}^0,\mathbf{l}^0)$ as the *initial configuration* of S. For simplicity we will often suppress the dependence on ω.

Let $T_p(n)$ be the time at which the n-th token is generated at station p and let $D_p(t)$ be the number of tokens generated up to time t at station p. These do not count the tokens present at time 0. In the rest of the paper we investigate structural properties of the *token generation* process $\mathbf{D} = (D_1(t), D_2(t), \ldots, D_m(t))$.

Remark 1.1 (Tandem GBS Networks) An open tandem network of GBS stations can be obtained as a special case of a cyclic network by considering an infinite buffer with an infinite number of jobs and holes between server m and server 1, *i.e.*, by letting $k_1 = \infty, j_1 = \infty$, and $k_1 - j_1 = \infty$. By properly handling ∞ in various equations and inequalities it can be verified that the above model reduces to an open tandem network. The results of Section 6 on the symmetry of the throughput in number of jobs make sense only when $K < \infty$. However, all of the intermediate structural results on duality, reversibilty and concavity hold for open tandems of queues.

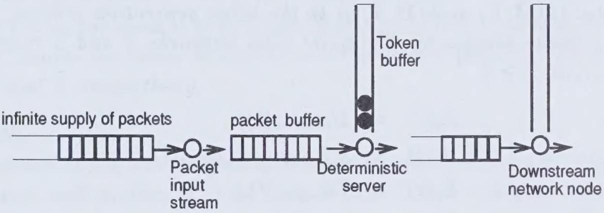


Figure 2: GBS tandem network for study of Leaky Bucket mechanism.

Example 1.2 (Leaky Bucket Mechanism) The leaky bucket mechanism has been studied by a number of authors (see Berger [4], Berger and Whitt [5], and references therein), as a mechanism to regulate the input rate of packets into high speed networks. A simple GBS tandem network that models such a leaky bucket mechanism is shown in Figure 2. Here a possibly bursty source feeds packets into the finite (or possibly infinite) packet buffer of the leaky bucket mechanism. The leaky bucket node is a GBS station with parameters $b = c$ being the token bucket size. There is a downstream network node with a finite or infinite job buffer. The principal aim of the mechanism is to reduce the burstiness of the input packet stream, while assuring a certain throughput. Several first and second order properties of this network that have been observed for the leaky bucket scheme are derivable from its analysis as a tandem of GBS stations. These include (1) the monotonicity of the throughput as a function of the service times at various stations, and of the job and token buffer sizes , (2) the convexity of the throughput as a function of the service times at various stations, and of the job and token buffer sizes, and (3) the invariance of the throughput on varying the job and token buffer sizes while maintaining their sum at a constant (see [6], [4], [5]).

2. Duality

A simple but important observation for networks of stations with communication blocking is that of *job–hole duality, i.e.*, any cycle or tandem of such stations can be identified with a dual network of holes flowing in the opposite direction. Duality was first introduced by Gordon and Newell [12], and has since played a critical role in establishing several results for networks with communication blocking (see for instance Ammar and Gershwin [2], Dallery and Towsley [8], Akyildiz and von Brand [1]). One of the key contributions of this paper is the recognition that general blocking of jobs is equivalent to the general starvation of holes. By incorporating general blocking and general starvation of jobs in the same network we ensure that the dual of a GBS network is another such network.

Definition 2.1 *Given any cyclic GBS network* $S = S((\mathbf{k}, \mathbf{b}, \mathbf{c}), (\mathbf{j}^0, \mathbf{l}^0), \omega)$, *we define its* dual *to be the cyclic GBS network* $\bar{S} = S((\bar{\mathbf{k}}, \bar{\mathbf{b}}, \bar{\mathbf{c}}), (\bar{\mathbf{j}}^0, \bar{\mathbf{l}}^0), \bar{\omega})$, *where* $\bar{k}_p = k_{m\oplus 2\ominus p}$, $\bar{b}_p = c_{m+1-p}$, $\bar{c}_p = b_{m+1-p}$, $\bar{j}^0_p = k_{m\oplus 2\ominus p} - j_{m\oplus 2\ominus p}$, $\bar{l}^0_p = l_{m+1-p}$, $\bar{\omega}_p = \omega_{m+1-p}$ *for* $1 \le p \le m$.

Observe that the dual of a network with general blocking alone, *i.e.*, a network with $\mathbf{c} = (0, \ldots, 0)$ is a network with general starvation alone, *i.e.*, a network with $\mathbf{b} = (0, \ldots, 0)$ and *vice versa.* The next lemma establishes the precise connection between the dynamics of a cyclic GBS network and its dual.

Lemma 2.2 *Let* $(\mathbf{D}, \mathbf{J}, \mathbf{L})$ *and* $(\bar{\mathbf{D}}, \bar{\mathbf{J}}, \bar{\mathbf{L}})$ *be the token generation process, job queue process, and token queue process for the cyclic GBS networks* S *and* $\bar{S}$ *respectively. Then for all* $p = 1, \ldots, m$, $t \ge 0$,

$$\begin{aligned} \bar{D}_p(t) &= D_{m+1-p}(t), \\ \bar{J}_p(t) &= k_{m\oplus 2\ominus p} - J_{m\oplus 2\ominus p}(t), \\ \bar{L}_p(t) &= L_{m+1-p}(t). \end{aligned} \tag{2.1}$$

Proof: Once we observe that $\bar{S}$ is the cyclic network of holes that flow against the flow of jobs in S, it is easy to see that whenever a token in generated in station p of S, another is generated in station $m + 1 - p$ of $\bar{S}$. The proof can be made precise by an induction on time. □

3. The Constraint Matrix

The token generation process $\mathbf{D}(t)$ of a cyclic GBS network, S, is obtained in an apparently complex manner from the GBS parameters $(\mathbf{k}, \mathbf{b}, \mathbf{c})$, the initial configuration $(\mathbf{j}^0, \mathbf{l}^0)$ and and the clock time sequence ω. The key result obtained in this section reduces this apparent complexity by asserting that the token generation process depends on $(\mathbf{k}, \mathbf{b}, \mathbf{c})$ and $(\mathbf{j}^0, \mathbf{l}^0)$ only through a matrix, $\mathbf{U} = \{u_{p,q} : 1 \le p, q \le m\}$, of non-negative integers. This matrix, called the *constraint matrix* of S is first defined below.

For $1 \le p, q \le m$, define

$$y_{p,q} := l^0_p + \sum_{i=p\oplus 1}^{q} j^0_i - l^0_q, \tag{3.1}$$

$$u_{p,q} := \begin{cases} [(c_p + J) \wedge (b_p + \sum_{i=p\oplus 1}^{q} k_i)] - y_{p,q} & \text{if } p \ne q \\ 0 & \text{if } p = q. \end{cases} \tag{3.2}$$

Proposition 3.1 *Let $S = ((\mathbf{k}, \mathbf{b}, \mathbf{c}), (\mathbf{j}^0, \mathbf{l}^0), \omega)$ be a GBS cyclic network with constraint matrix $\mathbf{U}$. The server at station p of S is disabled at time t, iff $D_p(t) = D_q(t) + u_{pq}$ for some $1 \leq q \neq p \leq m$.*

Proof: See [15].

4. Reversibility

Given a cyclic or tandem network, its *reversed* network has traditionally been defined as the network obtained by reversing the direction of flow of jobs. Moreover, the network is said to be *reversible* if the reversed and original networks have the same throughput (see Buzacott and Shanthikumar [7]). Following the approach of Muth [14], several researchers (see [17], [13], [8], [7], [9]) have established reversibility for several classes of tandem, cyclic, as well as fork-join networks. In this section we identify reversed networks for certain cyclic (or tandem) GBS networks and show that the reversed and original networks have the same throughput.

Theorem 4.1 *Let S and $\hat{S}$ be cyclic GBS networks with constraint matrices $\mathbf{U}$ and $\hat{\mathbf{U}}$ respectively, such that*

$$u_{p,q} = \hat{u}_{m+1-q,m+1-p}. \tag{4.1}$$

Let the service time processes, ω and $\hat{\omega}$ of the two systems, be related by $\omega_p^i = \hat{\omega}_{m+1-p}^{n+1-i}$, $1 \leq p \leq m$, $1 \leq i \leq n$, for some $n \geq 0$. Then, $\max_{1 \leq p \leq m} T_p(n) = \max_{1 \leq p \leq m} \hat{T}_p(n)$, where $T_p(n)$ and $\hat{T}_p(n)$ denote the times at which the n-th token is generated at station p under the networks S and $\hat{S}$, respectively.

Proof: See [15].

As a consequence of the above theorem, we can conclude that if the random sequence of service times at each station is i.i.d., then S and $\hat{S}$ have the same throughput. We need a prescription to obtain the network $\hat{S}$ whose constraint matrix $\hat{\mathbf{U}}$ satisfies (4.1) in order to apply Theorem 4.1. We say that a cyclic GBS network S is *reversible* iff $k_p - b_p + b_{p\ominus 1} + c_p - c_{p\ominus 1} \geq 1$, $1 \leq p \leq m$.

Proposition 4.2 *Given any reversible cyclic GBS network $S = S((\mathbf{k}, \mathbf{b}, \mathbf{c}), (\mathbf{j}^0, \mathbf{l}^0))$ there exists a network $\hat{S} = S((\hat{\mathbf{k}}, \hat{\mathbf{b}}, \hat{\mathbf{c}}), (\hat{\mathbf{j}}^0, \hat{\mathbf{l}}^0))$ whose constraint matrix $\hat{\mathbf{U}}$ satisfies (4.1). Moreover, $\hat{S}$ satisfies*

$$\hat{k}_{m\oplus 2\ominus p} := k_p - b_p + b_{p\ominus 1} + c_p - c_{p\ominus 1}, \tag{4.2}$$

$$\hat{b}_{m+1-p} := b_p, \tag{4.3}$$

$$\hat{c}_{m+1-p} := c_p. \tag{4.4}$$

$$\hat{J} = \sum_{p=1}^{m} \hat{j}_p^0 = J \tag{4.5}$$

Note that we have not specified the initial configuration of $\hat{S}$. For a complete specification of $\hat{S}$ and a proof of the above result, we refer the reader once again to [15]. A network $\hat{S}$ satisfying (4.2)–(4.5), and whose constraint matrix $\hat{\mathbf{U}}$ satisfies (4.1), is said to be the *reversed network* of S.

5. Concavity

In this section we investigate concavity of the token generation process as a function of the various parameters of a cyclic GBS network. The results of this section are based on a pathwise comparison of one GBS network with a convex combination of others using a mechanism called *synchronization.* Note that when the clock times for each event are i.i.d., exponential, as well as independent across the various events, then synchronization corresponds to uniformization. In Section 4, we saw that the token generation process $\mathbf{D}$ of a cyclic GBS network are dependent on its GBS parameters and the initial configuration, only through the constraint matrix $\mathbf{U}$. Below, we explore the nature of the relationship between the token generation process $\mathbf{D}$ and the constraint matrix $\mathbf{U}$.

Definition 5.1 Let $\Delta \subset \mathbb{Z}^q$ and let $\mathbf{f} : \Delta \to \mathbb{Z}^r$ be a function. We say that $\mathbf{f}$ is *s-concave* (resp. *s-increasing-concave*) iff for any $\mathbf{u}^0 = \frac{1}{s}\sum_{i=1}^{s} \mathbf{u}^i$ (resp. $\mathbf{u}^0 \geq \frac{1}{s}\sum_{i=1}^{s} \mathbf{u}^i$)

$$\mathbf{f}(\mathbf{u}^0) \geq \frac{1}{s}\sum_{i=0}^{s} \mathbf{f}(\mathbf{u}^i), \tag{5.1}$$

where $\mathbf{u}^i \in \Delta$ for all i. If f is s-concave (s-increasing-concave) for all $s \geq 1$ then we say that f is *concave* (*increasing-concave*).

Theorem 5.2 *The token generation process,* $\mathbf{D}(t,\omega)$*, in a synchronized family of cyclic GBS networks is an increasing-concave function of* $\mathbf{U}$ *for all* $t \geq 0$*,* ω*.*

Proof: See [15].

Corollary 5.3 *The token generation process,* $\mathbf{D}(t,\omega)$*, in a synchronized family of cyclic GBS networks is concave with respect to the initial configuration* $(\mathbf{j}^0, \mathbf{l}^0)$*, and is increasing-concave with respect to the GBS parameters* $(\mathbf{k},\mathbf{b},\mathbf{c})$ *for all* $t \geq 0$*,* ω*.*

Proof: By the previous theorem, the token generation process, $\mathbf{D}$, is increasing-concave with respect to $\mathbf{U}$. Moreover, $\mathbf{U}$ is concave with respect to the initial configuration, and is increasing-concave as a function of the GBS parameters. The result follows, if we note that the composition of increasing-concave functions is increasing-concave, and that the composition of an increasing-concave function with a concave function is concave. ☐

6. Structural Properties of the Throughput, $\Lambda(J)$

In this section we use the results of previous sections to derive some structural properties of the throughput as a function of the number of jobs J in a cyclic network with a fixed set of GBS parameters $(\mathbf{k},\mathbf{b},\mathbf{c})$. The throughput, Λ, of the network is defined by

$$\Lambda := \lim_{n\to\infty} \frac{n}{T(n)}$$

(whenever the limit exists), where $T(n) := \max_{1\leq p\leq m} T_p(n)$ is the first time that all the stations have generated at least n tokens. Throughout this section we will assume the service time sequence is a stochastic process such that almost surely the limit in the definition of the throughput exists and is a constant that depends on the initial configuration $(\mathbf{j}^0, \mathbf{l}^0)$ only through the total number of customers J. When using reversibility, we will make the further assumption that the service times are i.i.d. and independent across stations. When using concavity, we will additionally assume that the service times are exponential. The exponential service time assumption can in fact be relaxed to *PERT* type distributions as is done in Baccelli and Liu [3].

We establish symmetry of the throughput $\Lambda(J)$ in the number of jobs J for various classes of cyclic GBS networks in Theorems 6.2, 6.5, and 6.10. In Corollaries 6.3, 6.6, and 6.11, we combine this with the concavity result of Section 7, to show that $\Lambda(J)$ is non-decreasing up to $J^* = K/2$ ($J^* = K/2 + \Delta$ for some Δ in Corollary 6.11) and non-increasing after that, and is thus maximized at J^*. The results in Theorem 6.2 and Corollary 6.3 generalize the results obtained by Dallery and Towsley [8] and in Buzacott and Shanthikumar [7] for cyclic networks with communication blocking and starvation (CBS) to a much larger class of *self-reverse-dual* GBS networks. The results in Theorem 6.10 and Corollary 6.11 generalize the results obtained by Buzacott and Shanthikumar [7] for 2-station networks with manufacturing blocking and exponential service times. An important feature of these results is that they use general starvation crucially even for networks with general blocking alone. Moreover, Theorem 6.10 establishes symmetry for i.i.d. service times with general distribution (not necessarily exponential). Finally, in Theorem 6.12 and its corollaries, we directly show that the throughput is non-decreasing in the regime of light usage (small J) and is non-increasing in the regime of heavy usage (large J). For a certain class of networks with $k_{\max} \geq K/2$, these regimes overlap and thus one can identify a range of J which maximize the throughput. The constraint $k_{\max} \geq K/2$ is automatically satisfied for 2-station networks, and thus Corollary 6.15 strengthens the throughput maximization result of Corollary 6.11.

Definition 6.1 *A cyclic GBS network S is* self-reverse-dual *if its GBS parameters* $(\mathbf{k}, \mathbf{b}, \mathbf{c})$ *satisfy the condition that* $\mathbf{b} = \mathbf{c}$.

It is easy to see that a self-reverse-dual network is reversible. Further, from Definition 2.1 and Proposition 4.2 we can deduce that the parameters $\mathbf{k}, \mathbf{b}, \mathbf{c}$ and J of a self-reverse-dual cyclic GBS network S are identical to those of the dual network of the reversed network, $\hat{\bar{S}}$, (and the reversed network of the dual network, $\bar{\hat{S}}$).

Theorem 6.2 *Let S be a self-reverse-dual network with GBS parameters* $(\mathbf{k}, \mathbf{b}, \mathbf{c})$ *and i.i.d. service times, independent across servers. Then the throughput is symmetric about $K/2$, i.e., $\Lambda(K-J) = \Lambda(J)$ for any $J = 0, \ldots, K$.*

Proof: This result follows easily by combining duality with reversibility. Pick any valid initial configuration $(\mathbf{j}^0, \mathbf{l}^0)$, and consider the cyclic GBS network $S = S((\mathbf{k}, \mathbf{b}, \mathbf{c}), (\mathbf{j}^0, \mathbf{l}^0))$. The condition $\mathbf{b} = \mathbf{c}$ for self-reverse-dual networks implies that the network is reversible, and that the reversed network of the dual network has $K - J$ jobs, but the same $(\mathbf{k}, \mathbf{b}, \mathbf{c})$ parameters as S. Moreover, by Lemma 2.2, Theorem 4.1 and the i.i.d. assumption, it follows that $T'(n) \overset{\text{st}}{=} T(n)$, $n \geq 1$. Thus, $\Lambda(K-J) = \Lambda(J)$. □

Corollary 6.3 *In addition to the assumptions in the above theorem, assume that the service times are exponential and that $K \geq m+1$. Then $\Lambda(J)$ is non-decreasing in $J \in \{0, \ldots, \lfloor K/2 \rfloor\}$, non-increasing in $J \in \{\lceil K/2 \rceil, \ldots, K\}$, and is maximized at $[K/2]$, where $[x]$ is (any of) the integer(s) closest to x.*

Proof: See [15].

Another family of cyclic GBS networks where we can deduce symmetry of throughput, is that of *self-dual* networks defined below:

Definition 6.4 *A cyclic GBS network $S((\mathbf{k},\mathbf{b},\mathbf{c}),(\mathbf{j}^0,\mathbf{l}^0),\omega)$ is* self-dual *iff the stations can be renumbered through a cyclic shift $i = 0, 1, \ldots, m-1$, such that the GBS parameters of the resulting network are identical to those of the dual network $\bar{S}$, i.e., $(\bar{k}_p, \bar{b}_p, \bar{c}_p) = (k_{p\oplus i}, b_{p\oplus i}, c_{p\oplus i})$, $1 \le p \le m$; and if the service time process of the resulting network is statistically identical to that of $\bar{S}$, i.e., $\{\bar{\omega}_p,\ 1 \le p \le m\} \overset{\text{st}}{=} \{\omega_{p\oplus i},\ 1 \le p \le m\}$*

Theorem 6.5 *Let S be a self-dual network with GBS parameters $(\mathbf{k},\mathbf{b},\mathbf{c})$. Then the throughput is symmetric about $K/2$, i.e., $\Lambda(K-J) = \Lambda(J)$ for any $J = 0, \ldots, K$.*

Proof: Follows from Lemma 2.2. □

Corollary 6.6 *In addition to the assumptions in the above theorem, assume that $K \ge m+1$, and that the service times are i.i.d. exponential, independent across stations. Then $\Lambda(J)$ is non-decreasing in $J \in \{0, \ldots, \lfloor K/2 \rfloor\}$, non-increasing in $J \in \{\lceil K/2 \rceil, \ldots, K\}$, and is maximized at $[K/2]$, where $[x]$ is (any of) the integer(s) closest to x.*

Proof: This proof is identical to that of Corollary 6.3. □

The above results can be further extended for the special case of 2-station cyclic networks based on the following theorem.

Theorem 6.7 *Let $S = S((\mathbf{k},\mathbf{b},\mathbf{c}),(\mathbf{j}^0,\mathbf{l}^0))$ and $S' = S((\mathbf{k}',\mathbf{b}',\mathbf{c}'),(\mathbf{j}^{0\prime},\mathbf{l}^{0\prime}))$ be 2-station cyclic GBS networks such that $\mathbf{k}' = \mathbf{k}$, and $(b'_p - b_p) = -(c'_q - c_q) = (y'_{p,q} - y_{p,q})$ for $p, q = 1, 2; p \ne q$. Then S and S' have the same token generation process.*

Proof: See [15].

Definition 6.8 *A 2-station cyclic GBS network $S = S((\mathbf{k},\mathbf{b},\mathbf{c}),(\mathbf{j}^0,\mathbf{l}^0))$ is* switchable *if $b_p - c_p = \Delta$, $p = 1, 2$ and $2(\Delta \vee 0) \le J \le K + 2(\Delta \wedge 0)$.*

Proposition 6.9 *If S is a 2-station switchable cyclic GBS network, then there exists another 2-station cyclic GBS network $\ddot{S} = S((\ddot{\mathbf{k}},\ddot{\mathbf{b}},\ddot{\mathbf{c}}),(\ddot{\mathbf{j}}^0,\ddot{\mathbf{l}}^0))$, called the* switched network *of S, such that $\ddot{\mathbf{k}} = \mathbf{k}$, $\ddot{\mathbf{b}} = \mathbf{c}$, $\ddot{\mathbf{c}} = \mathbf{b}$, $\ddot{J} = J - 2\Delta$, and $(\ddot{b}_p - b_p) = -(\ddot{c}_q - c_q) = (\ddot{y}_{p,q} - y_{p,q})$ for $p, q = 1, 2; p \ne q$. Moreover, $\ddot{S}$ has the same throughput as S.*

Proof: See [15].

Theorem 6.10 *Let S be a 2-station cyclic network with GBS parameters $(\mathbf{k},\mathbf{b},\mathbf{c})$ such that $b_p - c_p = \Delta$, $p = 1, 2$. Assume that the service time process is i.i.d. and independent across servers. Then the throughput is symmetric about $K/2 + \Delta$, i.e., $\Lambda(K + 2\Delta - J) = \Lambda(J)$ for all $2(\Delta \vee 0) \le J \le K + 2(\Delta \wedge 0)$.*

Proof: This result follows easily by combining switchability, duality and reversibility. See [15] for details. □

This result generalizes the result in Buzacott and Shanthikumar [7] for manufacturing blocking, *i.e.*, $b_p = 1$, $c_p = 0$, $p = 1, 2$, and exponential service times. Our proof technique does not rely on explicit calculation of the stationary distribution and the throughput as is done in [7]. Instead, it makes use of switchability, duality, and reversibility. A key point here is that we need to make use of general starvation as a dual of general blocking to prove the above result even for networks with general blocking alone.

Corollary 6.11 *In addition to the assumptions in the above theorem, assume that the service times are exponential and that* $K \geq m+1$. *Then* $\Lambda(J)$ *is non-decreasing in* $J \in \{2(\Delta \vee 0), \ldots, \lfloor K/2 \rfloor + \Delta\}$, *non-increasing in* $J \in \{\lceil K/2 \rceil + \Delta, \ldots, K + 2(\Delta \wedge 0)\}$, *and is maximized at* $[K/2] + \Delta$, *where* $[x]$ *is (any of) the integer(s) closest to* x.

Proof: This proof is identical to that of Corollary 6.3. □

This result is strengthened in Corollary 6.15 by relaxing the condition that the service time process be i.i.d. exponential. In order to do so we replace the use of concavity with a monotonicity result.

In the following theorem we extend Corollary 3.13 of Shanthikumar and Yao [16] to derive regimes where Λ is monotonic in the number of jobs in cyclic GBS networks.

Theorem 6.12 *Let* $S = S((\mathbf{k}, \mathbf{b}, \mathbf{c}), (\mathbf{j}^0, \mathbf{l}^0))$ *be a cyclic GBS network.*

- **Light Usage**: *Let* r *be a station such that*

$$k_r > J + c_p - b_p - \sum_{s=p\oplus 1}^{r\ominus 1} k_s, \quad \forall p \neq r. \tag{6.1}$$

Then adding a job to station r *can only increase the token generation process.*

- **Heavy Usage**: *Let* r *be a station such that*

$$k_r > K - J + b_p - c_p - \sum_{s=r\oplus 1}^{p} k_s, \quad \forall p \neq r \ominus 1. \tag{6.2}$$

Then removing a job from station r *can only increase the token generation process.*

Proof: See [15].

Corollary 6.13 *Let* S *be a cyclic network with GBS parameters* $(\mathbf{k}, \mathbf{b}, \mathbf{c})$ *such that* $b_p - c_p = \Delta,\ p = 1, \ldots, m$. *Then the throughput* $\Lambda(J)$ *is non-decreasing in* $J \in \{0, \ldots, k_{\max} + \Delta\}$ *and non-increasing in* $J \in \{K - k_{\max} + \Delta, \ldots, K\}$, *where* $k_{\max} = \max\{k_p : 1 \leq p \leq m\}$.

Proof: Let r be a station such that $k_r = k_{\max}$. Assume that $J < k_{\max} + \Delta = k_r + \Delta$. By the condition $b_p - c_p = \Delta,\ p = 1, \ldots, m$, it follows that $k_r > J - \Delta = J + c_p - b_p \geq J + c_p - b_p - \sum_{s=p\oplus 1}^{r\ominus 1} k_s,\ \forall p \neq r$. Thus (6.1) holds and by Theorem 6.12 it follows that adding a job can only increase the throughput. The proof for the decreasing part follows by duality. □

Corollary 6.14 *Let* S *be a cyclic network with GBS parameters* $(\mathbf{k}, \mathbf{b}, \mathbf{c})$ *such that* $b_p - c_p = \Delta,\ p = 1, \ldots, m$, *and* $k_{\max} = \max\{k_p : 1 \leq p \leq m\} \geq K/2$. *Then the throughput* $\Lambda(J)$ *is maximized by any* $J \in \{K - k_{\max} + \Delta, \ldots, k_{\max} + \Delta\}$.

Proof: Follows immediately from Corollary 6.13. □

Corollary 6.15 *Let* S *be a 2-station cyclic network with GBS parameters* $(\mathbf{k}, \mathbf{b}, \mathbf{c})$ *such that* $b_p - c_p = \Delta,\ p = 1, \ldots, m$. *Then the throughput* $\Lambda(J)$ *is maximized by any* $J \in \{K - k_{\max} + \Delta, \ldots, k_{\max} + \Delta\}$, *where* $k_{\max} = \max\{k_1, k_2\}$.

Proof: For a 2-station cyclic network, one always has $k_{\max} \geq K/2$. The result now follows immediately from Corollary 6.13. □

This result generalizes the result in Buzacott and Shanthikumar [7] for manufacturing blocking, *i.e.*, $b_p = 1,\ c_p = 0,\ p = 1, 2$, and i.i.d. exponential service times independent across servers. Our proof technique does not rely on explicit calculation of the stationary distribution and the throughput as is done in [7].

References

[1] I. F. Akyildiz and Horst von Brand. Dual and selfdual networks of queues with rejection blocking. *Computing*, 43:1–11, 1989.

[2] M. A. Ammar and S. B. Gershwin. Equivalence relations in queueing models of fork/join networks with blocking. *Performance Evaluation*, 10:233–245, 1989.

[3] F. Baccelli and Z. Liu. Comparison properties of stochastic decision free Petri nets. *IEEE Trans. Automat. Contr.*, 37:1905–1920, 1992.

[4] A. W. Berger. Performance analysis of a rate-control throttle where tokens and jobs queue. *IEEE J. Selected Areas in Comm.*, 9:165–170, 1991.

[5] A. W. Berger and W. Whitt. The impact of a job buffer in a token-bank rate-control throttle. *Comm. Stat.*, 8(4):685–717, 1992.

[6] K. C. Budka. Stochastic monotonicity and concavity properties of rate based flow control mechanisms. *IEEE Trans. Aut. Control*, to appear, 1992.

[7] J. A. Buzacott and J. G. Shanthikumar. *Stochastic Models of Manufacturing Systems.* Prentice-Hall, Englewood Cliffs, NJ, 1992.

[8] Y. Dallery and D. Towsley. Symmetry property of the throughput in closed tandem queueing networks with finite buffers. *Operations Research Letters*, 10:541–547, 1991.

[9] P. Glasserman and D. D. Yao. Structured buffer-allocation problems in production lines. preprint, June 1991.

[10] P. Glasserman and D. D. Yao. Monotonicity in generalized semi-Markov processes. *Math. Oper. Res.*, 17:1–21, 1992.

[11] P. W. Glynn. A GSMP formalism for discrete-event systems. *Proc. of IEEE*, 77:14–23, 1989.

[12] W. J. Gordon and G. F. Newell. Cyclic queueing systems with restricted length queues. *Operations Research*, 15:266–277, 1967.

[13] B. Melamed. A note on the reversibility and duality of some tandem blocking queueing systems. *Management Science*, 32:1648–1650, 1986.

[14] E. Muth. The reversibility property of production lines. *Management Science*, 25:152–158, 1979.

[15] R. Rajan and R. Agrawal. Cyclic networks with general blocking and starvation. To be published in *QUESTA*.

[16] J. G. Shanthikumar and D. D. Yao. Monotonicity and concavity properties in cyclic queueing networks with finite buffers. In Harry G. Perros and Tayfur Altiok, editors, *Queueing networks with blocking : proceedings of the first international workshop held in Raleigh, North Carolina, May 20-21, 1988*, pages 325–344. North-Holland, 1989.

[17] G. Yamazaki, T. Kawashiha, and H. Sakasegawa. Reversibility of tandem blocking queueing systems. *Management Science*, 31:78–83, 1985.

Adaptive Load Balancing in Networks

Erol Gelenbe and Xiaowen Mang
Department of Electrical Engineering, Duke University,
Durham, N.C. 27706, USA

Abstract

We propose and study the ELB (Equitable Load Balancing) algorithm for networks. It is a computationally efficient dynamic flow assignment algorithm of time complexity $O(lMN)$ where l is the number of links, M is the number of active source-destination pairs and N is the number of nodes. It adaptively modifies routes and the amount of traffic allocated to each link as user traffic patterns fluctuate, so as to reduce congestion and distribute equitably traffic equitably.Simulations show that ELB adapts rapidly to traffic changes, and that it provides substantial improvements in end-to-end delays, compared to fixed single or multiple route schemes.

1 Introduction

One of the major challenges of future high-speed integrated services networks will be to provide an optimum cost-performance trade-off to users having widely varying needs and constraints. These needs and constraints, expressed in terms of bandwidth, performance metrics and cost, will vary among users, and will also vary over time for a particular user. In this paper we design an adaptive routing policy which we call ELB (Equitable Load Balancing) for high-speed communication networks, whose objective is to achieve on-line traffic distribution among links in a manner which is proportional to the link capacities. Its purpose is to insure that at all times congestion is reduced by an apropriate distribution of traffic in the network. ELB is based on a cost function representation of the load-balancing criterion in order to make decisions concerning traffic routing. The algorithm uses gradient-descent of the cost function in order to update the routing tables.

We first review some of the literature on network routing and control. We then describe the Equitable Load Balancing algorithm is designed. Simulation results are presented to illustrate how network performance is improved by ELB when traffic fluctuates.

Several authors, including [3] [4] have discussed the projected evolution of communication and computer-communication networks. Networks are moving towards Broadband ISDN (B-ISDN) based on ATM (Asynchronous Transmission Mode) technology, and network control is expected to evolve towards "Advanced Intelligent Networks (IN)" in which the service control logic would be separated from the connection control logic to enhance flexibility. A complete survey of routing and congestion control in networks would cover several hundred references. Therefore we only concentrate on a few examples of this abundant literature concerning an area of considerable significance. In [7] an alternate routing policy was proposed as a new congestion control scheme based on rerouting a portion of the traffic from a congested primary path to a predefined alternate path. In

[8], Z. Wang and D.W. Browning reported a distributed optimal routing algorithm. A. Kershenbaum, P. Kermani, and G.A. Grover [9] have developed a heuristic optimal routing algorithm which uses utilization rather than delay as a figure of merit. G.R Ash, F. Chang and D. Medhi [10] have designed traffic restoration algorithms for dynamic routing networks to overcome the effects of link or node failure. J. Filipiak, P. Chemouil [12] present various routing architectures for ATM networks based on recent User/Network Interface (UNI) protocols. G.I. Stassinopoulos and M.G. Kazantazakis [14] present an optimal routing algorithm based on an iterative link-by-link optimization involve in a computational effort of $O(N^5)$ per iteration. In [16], L. Fratta, M. Gerla and L. Kleinrock have proposed the Flow Deviation (FD) method which was further investigated in [1]. A. Gersht and R. Weihmayer [17] have integrated the routing, topological design, capacity assignment and facility selection process, through a link reduction algorithm which preserves connectivity and reliability constraints. In [18], C. Kang and H.H. Tan used FD to solve for the joint optimization of capacity and flow assignment. The FD method has been applied to load balancing in distributed systems by E. De Souza, E. Silva, and M. Gerla [19]. Recently, J.W. Roberts and A. Gravey have presented a useful and broad review of B-ISDN/ATM traffic modelling and performance analysis [6].

We consider the use of increased network intelligence *to achieve on-line performance management* of the network. We study routing and flow control as a means of improving network performance in real-time, as traffic patterns and intensities fluctuate. An effective routing scheme should: provide good user performance, by yielding the lowest possible end-to-end delays for the desired traffic, manage the network in a balanced manner in order to preserve as much free link capacity as possible, so as to be able to meet new traffic requests as they arise, and achieve these goals using algorithms of lowest possible computational complexity in order to allow them to be implemented in real-time, It should also be robust to imperfect our out-of-date information. These goals can obviously be be contradictory, so that a good algorithm is one which achieves a reasonable compromise between these objectives.

2 Equitable Load Balancing

We consider a network composed of N "nodes", and "links". Each node, representing a network switch, has a small number of incoming links from, and outgoing links to other nodes. Traffic circulating in the network is in the form of "packets" for a packet-switching network, or "cells" which are the units of information used in ATM networks [6]. Each node has a buffer pool for the packets which are transiting through it. The capacity of each link determines the speed at which it can transmit packets. In the sequel we concentrate on a high-speed packet-switching network model where capacities and traffic rates are expressed in packets per unit time. The following notation will be used: (i,j) denotes the link from node i to node j, $c(i,j)$ is the link capacity, and $\lambda(i,j)$ denotes the traffic, in packets per unit time, on link (i,j). $r(i)$ is the total outgoing capacity, in packets per unit time, of node i. $\Lambda(s,d)$ is the user traffic sent from source node s to sink d. $\delta(i,j,s,d)$ is the "pass permit"; when $\delta(i,j,s,d)$=1, traffic on route (s,d) can travel over link (i,j), otherwise $\delta(i,j,s,d)$=0. $\delta(i,j,s,d)$ is initially fixed, and will not be changed during updates of the ELB algorithm. $R(i,j,s,d)$ is the proportion of packets with source address s and destination address d traveling over link (i,j). These values can be interpreted in probabilistic terms. In particular, we will use $p(i,j)$ to denote the probability that a packet leaving node i goes to node j, and $d(i)$ for the departure probability from node i to the outside world. Traffic traveling over the path (s,d) will use link (i,j) if this link has been assigned to the path, i.e. if $c(i,j) \neq 0$ and $\delta(i,j,s,d) = 1$. We suppose that $\delta(i,j,s,d)$ is precomputed by some known route selection algorithm which respects the following constraints: End-to-end connectivity must be guaranteed for every source-to-sink pair, and the flow from any source to any destination is loop-free. The inequality $\lambda(i,j) < c(i,j)$ must also be satisfied for all (i,j) to guarantee that links and nodes will not saturate.

In the ELB policy, when a request for a new source-to-sink communication path (s,d) is made by some user, first a set of available intermediate links (i,j) are selected. This selection is made using a "shortest" path algorithm which takes into account issues such as communication delays and reliability. Then, the traffic assignment on each link (i,j) for this (s,d) pair, denoted by $R(i,j,s,d)$, is computed adaptively and on-line, on the basis of a cost function which reflects the ELB objective. We assume that users declare their needs in terms of an end-to-end traffic rate and connectivity. Our adaptive algorithm then provides the network with on-line modification of routes and link traffic so as to minimize a relevant cost function, while satisfying users' demands. The term "on-line" means that at selected instants t_k, $k = 1, 2, ...$, the algorithm gracefully and progressively modifies packet routing as a function of available information. These selected instants are those at which a change in the input traffic patterns is either declared (i.e. by the users), or detected by the system.

Let $\mathbf{\Lambda} = \{\Lambda(s,d)\}$ be the matrix of end-to-end traffic. We represent the time-varying traffic patterns by successive values of the traffic matrix $\mathbf{\Lambda_k}$, $k = 1, 2, ...$. The index value k refers to the k-th successive value taken by the traffic matrix, and not to the k-th time instant. Clearly for the controls to be effective, the time intervals t_k - t_{k-1}, $k = 1, 2, ...$ should be sufficiently large for the control algorithm to run, and for the network to reach a meaningful state before the traffic patterns vary once again. Thus, as with any adaptive control scheme, we will be concerned with "slowly varying" changes, where "slow" means slower than the network's internal time constants (including the control algorithm). The ELB algorithm's aim is to distribute load proportionally to the load carrying capacity of each link. The cost function which expresses this objective is given below:

$$K = \sum_i \sum_j \frac{1}{2}[p(i,j) - \frac{c(i,j)}{r(i)}]^2 \qquad (1)$$

Though a quadratic cost function has been chosen, some other appropriate differentiable function could also have been taken to express ELB. The ratio $c(i,j)/r(i)$ expresses the load carrying capacity of link (i,j) with respect to the output capacity of node i; $p(i,j)$, the proportion of traffic leaving node i which takes link (i,j), is the quantity which will be modified by the control; i.e. $p(i,j)$ will be modified by the routing decision R(i,j,s,d). K expresses the objective that the proportion of traffic flowing over link (i,j) $p(i,j)$, must be as close as possible to the relative link capacity (c(i,j)/r(i)). Thus K expresses the squared deviation over the network, from an ideal distribution of traffic.

The ELB routing policy establishes and modifies on-line traffic routing according to the following procedure:

Given: Topology, link capacities, pass permit matrix $\delta = \{\delta(i,j,s,d)\}$ at nodes and communication requirement matrix $\mathbf{\Lambda}$,
Minimize: the cost metric K,
Using the: routing decision R(i, j, s, d)
Subject to the following constraints: (i) **if** $\delta(i,j,s,d) = 0$, **then** $R(i,j,s,d) = 0$
(ii) **if** $c(i,j) = 0$

then $R(i,j,s,d) = 0$ and $p(i,j) = 0$
(iii) $0 \leq R(i,j,s,d) \leq 1$
if $\sum_k^N R(k,i,s,d) = 0$
then $\sum_j^N R(i,j,s,d) = 0$
else $\sum_j^N R(i,j,s,d) = 1$
$k,i,j,s,d \in 1,...,N$

(i) and (ii) imply that route selection (and hence flow assignment) are based on topology and available capacity of the network; (iii) guarantees that whenever a packet traveling from source s to destination d arrives to node i, it is sure to be sent to one of the nodes downstream of j (or to the outside world). In the construction of the adaptive policy we compute the gradient of the cost function with respect to the control parameter R. In the following, we will use (S,D) to denote a specific source-destination pair, while (s,d) will be a generic (i.e. any) such pair. From (1) we have $\frac{\partial K}{\partial R(u,v,S,D)} = \sum_i \sum_j \sigma[p(i,j) - \frac{c(i,j)}{r(i)}] \frac{\partial p(i,j)}{\partial R(u,v,S,D)}$, where $p(i,j)$ is the probability that a packet leaving node i heads for node j along link (i,j), Λ(s,d) is the rate of traffic sent from source node s to destination node d, and R(i,j,s,d) is the routing decision probability that a packet with source address s and destination address d travels over link (i,j). After some algebra we have:

$$\frac{\partial p(i,j)}{\partial R(u,v,S,D)} = G(i,j,u,v,S,D) - p(i,j)H(i,j,u,v,S,D) \tag{3}$$

$$G(i,j,u,v,S,D) = \begin{cases} 1 & \text{if i=u, j=v, s=S, d=D,} \\ \frac{R(i,j,s,d)}{R(u,v,s,d)} & \text{if s=S, d=D, and (u,v) is on (S,D) from S to i,} \end{cases}$$

$$H(i,j,u,v,S,D) = \left\{ \frac{1}{R(u,v,s,d)} \quad \text{if s=S, d=D, and (u,v) is on (S,D) from S to i,} \right.$$

and they are 0 for all other cases.

The ELB algorithm will adapt the network by computing successive values, for k=1, 2, ..., of the routing decision matrix $\mathrm{R}_k = \{\mathrm{R}_k(\mathrm{i, j, s, d})\}$ in response to traffic fluctuations in an attempt to minimize the cost function K_k. The algorithm will always *reduce* the cost function K, though it will not attain an absolute minimum. For a set of parameter values it will reach a *local minimum*, within some acceptable error bound used to stop the gradient descent iteration. For each new input traffic pattern $\mathbf{\Lambda}_k$, the k-th value of the decision matrix R_k will be computed based on gradient descent of the cost function K_{k-1}. For the source-destination pair (s,d) the rule for updating R_k is the well-known *gradient descent* rule which guarantees that at each iteration the cost function is reduced. First set $R_k \Leftarrow R_{k-1}$, and the iterate as follows until the cost function K_k computed between two successive iteration steps is less than ϵ :

$$R_k(u,v,S,D) \Leftarrow R_k(u,v,S,D) - \eta \frac{\partial K}{\partial R(u,v,S,D)}|_k \tag{4}$$

subject to the constraints expressed in Section 2. $\epsilon > 0$ is the stopping error for the iterations, and $\eta > 0$ is a constant which determines the speed of gradient descent. The following is a schematic description of the ELB algorithm for each successive value of k:

(a) Get new input values Λ_k and δ_k, and $R_k \Leftarrow R_{k-1}$.

(b) Use (2) to compute P_k. Compute $K_{k,old}$.

(c) Carry out one step of the iteration (4), using (3), to update R_k. Compute $K_{k,new}$. If $|[K]_{k,new} - [K]_{k,old}| < \epsilon$, then *stop*. Otherwise go back to (b).

Step (c), repeated until the stopping condition is satisfied, is the most computationally intensive portion of the algorithm. Since the R(i,j,s,d) depend on the path from s to d, the order in which the R(i,j,s,d) are updated must be chosen carefully subject to the constraints.

Each time we select a source-to-sink pair (S, D) where $\Lambda(S, D) \neq 0$, the network update is begun from the source node S. When the routing updates at some current node u are terminated, a successor node v is selected among its downstream nodes which satisfy the following conditions: the link (u, v) is part of the network, and all neighboring *upstream* nodes of v have already been updated in the current update iteration.

The choice of a "large" gradient descent speed η can accelerate the computation, but it may also lead to oscillations around a local minimum or around several local minima, while a small value may slow down the algorithm We suggest the use of an *adaptive gradient descent speed*, where η decreases gradually as long as the condition $K_{k,old} - K_{k,new} > \epsilon$ is met during the computation. This approach has yielded satisfactory results in the numerous examples we have considered.

An important issue for any network control algorithm is its computational complexity. In any centralized adaptive control algorithm which affects all the nodes of a network there are two important sources of overhead: (I) The portion of the algorithm which collects information *from* each node, and then which provides control information *to* the nodes, and (II) that part of the algorithm which actually carries out the computations of the new routing values $R(i, j, s, d)$. Clearly (I) is an inherent part of any centralized algorithm, and it requires messages to travel to and from all the nodes of the network to the centralized decision making site. This is a basic disadvantage of any centralized approach. Little can be done about this, except to take advantage of the network's topology to reduce the delays and the number of messages. Since here we consider a centralized control algorithm, we will suppose that this overhead is acceptable, and concentrate on evaluating the computational effort in (II). The algorithm's computational complexity resides in steps (b) and (c) of Section 2.2. For each source-to-sink pair (s,d) and for each link (i,j) as well as for each network input traffic pattern (successive k) we have to use equations (2), (3) and then (4), and (2) has to be solved once for each iteration of the algorithm. Notice that the complexity of computing (2) depends on the network's topology. Let l be the number of links in the network, and E be the maximum number of alternate routes per source-to-sink pair. M is the number of active source-to-sink pairs, and L denotes the maximum number of links on a path. Since L$\leq$N-1, the computation of p(i,j) from (2) is of time complexity no more than $O(EMN)$. In practical cases, E will be a small integer (say 2 or 3). Thus the complexity of computing (2) for *all* links will be *at worst* $O(MN^3)$, and in practice $O(lMN)$ since the number of links will in general be much smaller than N^2. The computation of any one step of the gradient descent is based on the derivative (3), which simply requires all the $p(i, j)$ to have been computed. This makes the ELB algorithm attractive, since its execution time is quite reasonable.

Simulation of the Control The simulation model we use is composed of two parts: the packet switching network simulator and the adaptive control simulator. These two parts interact via an interface (see Figure 2), which "triggers" the call to the adaptive algorithm when certain conditions occur. The trigger condition is a substantial change in the rate of arrivals of external traffic to the network, such as - for instance - the activation or deactivation of a source-destination pair. For the purpose of this evaluation, we assume that this change can either occur because:

(i) A new (s,d) communication path is explicitly requested by some user.

(ii) Or, there is a substantial change in traffic on a (s, d) pair, and this change is detected at the entry point. In the simulator a simple traffic estimator is used to detect this change.

The simulator allows the representation of an arbitrary network topology. It can be used to represent either a "cell-based" ATM network, or an ordinary packet switching network. The set of nodes and links of the network are "declared" to the simulator at initialization of the simulation, as shown on Table 1 for the network of Figure 1. Here for each source-destination pair (shown horizontally), a 1 or 0 indicates that the corresponding link (shown vertically) can be used. The packet length distribution can be arbitrary and is also declared at initialization, though the default option is to assume that it is deterministic and fixed to 1. This default assumption is used, in particular, when the simulated network is cell-based. Each link has a buffer to store the packets which are waiting for being sent out on this link. Packets which belong to the source-destination pair (s,d) are routed at each node according to R(i,j,s,d) with a mean service time 1/c(i,j) for link (i,j). For each (s,d) the entering traffic forms a Poisson process of rate $\Lambda(s, d)$.

To adaptively control the network in on-line mode, the network's traffic is measured over successive small time intervals, and the traffic rates are estimated. This information is entered into the traffic matrix λ, and then used by the ELB algorithm to update the routing matrix R in order to minimize the cost K. The time interval for parameter estimation is sufficiently small so that the network may adapt rapidly.

An example network is shown in Figure 1. There are 4 active communication pairs: (1,8), (2,7), (6,3), (8,2).Incoming traffic of the source-destination (s,d) pairs is generated according to a Poisson process. When the network is initiated for each (s,d) pair there is a single route. Then throughout the simulation, the routes are updated according to the ELB algorithm which is allowed to introduce multiple alternate routes within a limited set.

Transient behaviour of the controlled network The simulations shown on Figures 3 and 4 concern a high-speed network where link capacities which are all equal to 2.4x10^5 packets/second. On Figure 3, traffic values change each 0.05 secs. Each gradient descent iteration is assumed to take 0.001 secs of computing time, and this additional delay is included in the simulation. We see that the ELB algorithm adapts rapidly to changes in traffic and that it effectively controls the network.

Steady-state behaviour On Figure 4 we present simulation results concerning the end-to-end delay performance of ELB. For the network of Figure 1 we take the four (s, d) pairs (1,8), (2,7), (6,3), (8,2), and assign to each the same traffic value $\Lambda/4$ so that the total traffic through the network is Λ packets/sec. We vary Λ as shown on Figure 3 and measure the average end-to-end packet delay. The simulation results shown were conducted for a total of $2x10^6$ packets passing through the network. This simulation experiment is conducted with three routing policies: **Fixed single route policy.** Heere packets which belong to the same (s, d) pair follow one fixed route. In order to get the best results we deliberately let the routes of different (s, d) be completely disjoint. These are the routes given in Table 2. **Equally weighted multi-route policy.** Packets belonging to the same (s, d) follow all allowable routes of the "pass-permit" matrix of Table 1, and selection is made with equal probability at the output of each node. **ELB**

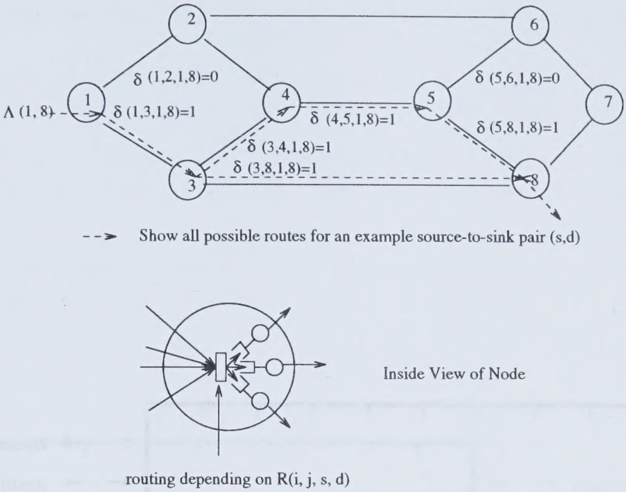

2
6
δ (1,2,1,8)=0
δ (5,6,1,8)=0
Λ (1, 8)
1
4
5
7
δ (1,3,1,8)=1
δ (4,5,1,8)=1
δ (5,8,1,8)=1
δ (3,4,1,8)=1
δ (3,8,1,8)=1
3
8
Show all possible routes for an example source-to-sink pair (s,d)
Inside View of Node
routing depending on R(i, j, s, d)

Figure 1: An Example Of a Network Topology

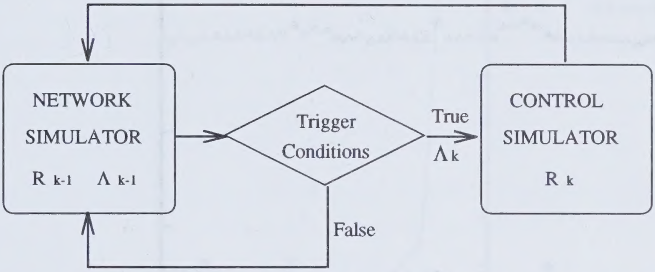

NETWORK
SIMULATOR
R k-1 Λ k-1
Trigger
Conditions
True
Λk
False
CONTROL
SIMULATOR
R k

Figure 2: Simulation Model

policy.

The simulations show that the fixed single route policy is better than fixed multi-route, because there is no interference between source-to-destination paths in the first case. However ELB is substantially better than either the other policies, since it dynamically optimizes the flow of traffic, yielding a substantial reduction in average end-to-end delays for the same traffic values. More examples could not be shown here because of page limitations.

References

[1] L. Kleinrock "Queueing Systems, Volume II: Computer Applications" *John Wiley & Sons, New York (1977).*

[2] D.P. Bertsekas "Data Networks" *Prentice-Hall, Englewood Cliffs, N.J. (1987).*

[3] H. Yunoki, S. Saito, J. Richards "Network Evolution and The Intelligent Network (IN)" *IEEE ICC'90, 221.1, 0192-0196, 1990.*

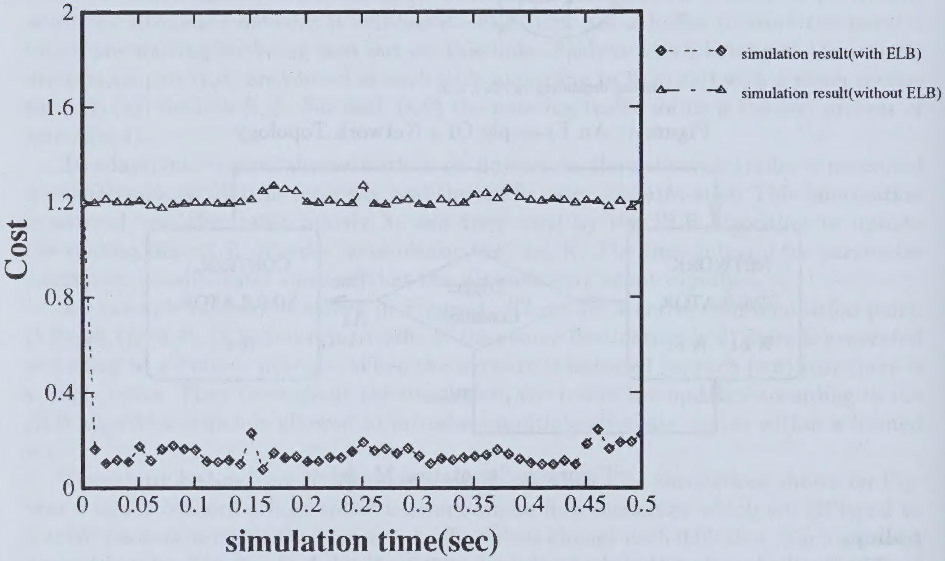


Figure 3: Simulation results for the network of Figure 1, without and with control of the ELB algorithm. All link capacities are $c(i,j) = 2.4x10^5$ packets/secs. For the (s,d) pairs (1,8), (2,7), (6,3), (8,2) we have traffic rates, varying very rapidly between high and low values each 0.05 secs. ELB control responds very well. Traffic values in multiples of 10^5 packets/sec are: at t=0.00 : 2.0, 1.5, 2.5, 2.5; t=0.05: 0.5, 0.8, 1.0, 0.5; t=0.1: 2.0, 1.5, 2.5, 2.4; t=0.15: 0.5, 0.5, 0.3, 0.6; t=0.20: 2.0, 1.5, 2.5, 2.5; t=0.25: 0.5, 0.8, 1.0, 0.5; t=0.30: 2.0, 1.5, 2.5, 2.5; t=0.35: 0.5, 0.5, 0.3, 0.6; t=0.40: 2.0, 1.5, 2.5, 2.5; t=0.45: 0.5, 0.8, 1.0, 0.5; t=0.50: 2.0, 1.5, 2.5, 2.4 .

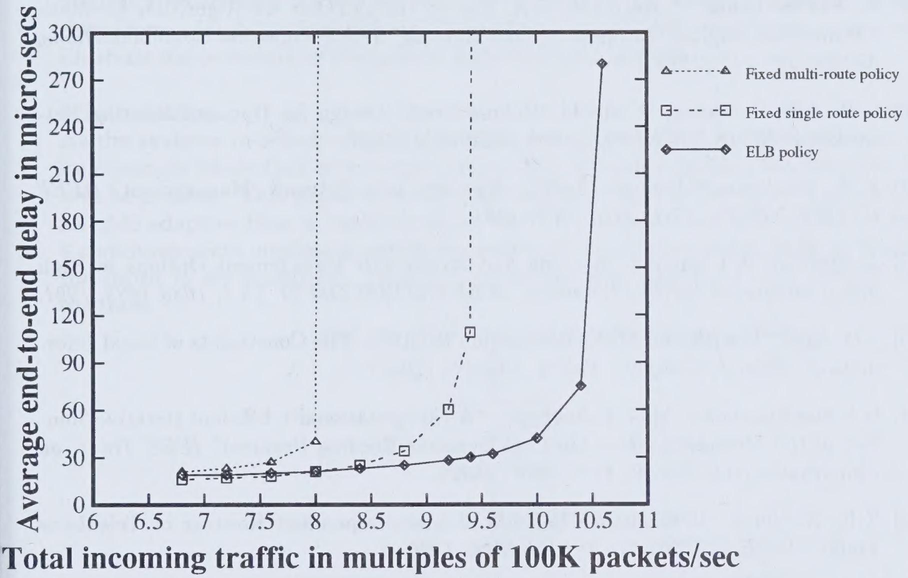


Figure 4: Comparison of ELB and two fixed routing policies, in terms of steady-state end-to-end delay versus total network load (incoming traffic). All link speeds are $c(i,j) = 2.4x10^5$ packets/sec. We see that ELB yields order of magnitude reductions in delay at high input traffic.

[4] R.A. Verstaete, H.J.M. Decuypere "Aspects of Intelligent Networks in an ATM enviroment" *IEEE ICC'90, 306B.1, 0434-0438, 1990.*

[5] R. Epley, B.A. Polobsky, S. Yeh "Advanced Intelligent Network Services Evolution" *IEEE ICC'90, 221.2, 0197-0204, 1990.*

[6] J.W. Roberts, A. Gravey "Recent Results on B-ISDN/ATM Traffic Modelling and Performance Analysis-A Review of ITC 13 Papers" *IEEE GLOBECOM'91, 37.3, 1325-1330, 1991.*

[7] P. Harshavardhana, S. Dravida, A.B. Bondi "Congestion Control for Connectionless Networks via Alternate Routing" *IEEE GLOBECOM'91, 11.3, 0339-0346, 1991.*

[8] Z. Wang, D.W. Browning "An Optimal Distributed Routing Algorithm" *IEEE Trans. on Communications, Vol.39, 1379-1387 (1991).*

[9] A. Kershenbaum, P. Kermani, G.A. Grover "MENTOR: An Algorithm for Mesh Network Topological Optimization and Routing" *IEEE Trans. on Communications, Vol.39, 503-512 (1991).*

[10] G.R. Ash, F. Chang, D. Medhi "Robust Traffic Design for Dynamic Routing Networks" *IEEE INFOCOM'91, 5D.1, 0508-0514, 1991.*

[11] K.R. Krishnan "Dynamic Traffic Routing and Network Management" *IEEE GLOBECOM'91, 37.7, 1346-1350, 1991.*

[12] J. Filipiak, P. Chemouil "Routing and Bandwidth Management Options in High Speed Integrated Services Networks" *IEEE GLOBECOM'91, 48.1, 1685-1689, 1991.*

[13] J.M. Jaffe "Distributed Multi-Destination Routing: The Constraints of Local Information" *Siam J. Comput., Vol.14, 875-888 (1985).*

[14] G.I. Stassinopoulos, M.G. Kazantzakis "A Computationally Efficient Iterative Solution of the Multidestination Optimal Dynamic Routing Problem" *IEEE Trans. on Communications, Vol.39, 1370-1378 (1991).*

[15] K.R. Krishnan "Performance Benefits of State-Dependent Routing of Telephone Traffic" *IEEE ICC'90, 334.3, 1314-1318, 1990.*

[16] L. Fratta, M. Gerla, L. Kleinrock "The Flow Deviation Method – An Approach to Store and Forward Communication Network Design" *Networks,* Vol.3, 97-133 (1973).

[17] A. Gersht, R. Weihmayer "Joint Optimization of Data Network Design and Facility Selection" *IEEE J. on Selected Areas in Comunications,* Vol.8, 1667-1681 (1990).

[18] C. Kang, H.H. Tan "Combined Channel Allocation and Routing Algorithms" *IEEE GLOBECOM'91,* 48.2, 1690-1695, 1991.

[19] E. De Souza, E. Silva, M. Gerla "Queueing Network Models for Load Balancing in Distributed Systems" *J. Parallel and Distributed Computing,* Vol.12, 24-38 (1991).

[20] E. Gelenbe "G-Networks with instantaneous customer movement", *Journal of Appl. Probability,* to appear (1993).

LMS Convergence and Performance With a Filtered-Error Update

by
John B. Kenney, Charles E. Rohrs and Vivek P. Telang
Tellabs Research Center
Mishawaka, IN 46545

Abstract- This paper is concerned with systems in which the output error of an LMS adaptive filter is subsequently filtered, e.g. a communication system with an echo canceller followed by an equalizer. Of interest is how the behavior of the LMS algorithm is affected if the filtered error is used to update the adaptive coefficients. The paper presents a theorem with proof setting out sufficient conditions for the deterministic convergence of the output error to zero. A second result is an analytical expression for the Misadjustment. Finally, simulations based on realistic data from a High-Bit Rate Digital Subscriber Line (HDSL) application are used to illustrate the potential advantages of a filtered-error update.

1. Introduction

As the systems in which adaptive filters are used become more sophisticated, it is increasingly likely that other signal processing elements besides the adaptive filter will be present. This paper is concerned with systems in which the output error of an LMS adaptive filter is itself subsequently filtered. An example of such a system is a communication modem in which the output of an adaptive echo canceller becomes the input to an equalizer. Figure 1 illustrates this concept with a simple block diagram.

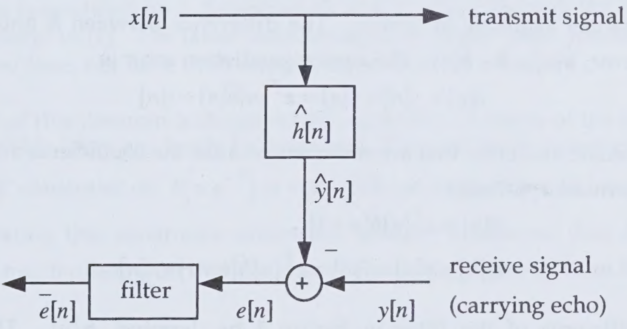


Figure 1: An adaptive echo canceller with filtered output error

While the LMS algorithm normally updates its adaptive coefficients using the output error, it may be possible to improve performance by using the filtered error instead. This paper examines the consequences for convergence and performance of an LMS algorithm employing a filtered-error update.

Following this introduction, the problem is described in more detail in Section 2. Section 3 presents the first result of the paper, a theorem setting out sufficient conditions for the deterministic convergence of the output error to zero. Stochastic analysis in Section 4 quantifies the Misadjustment of the filtered-error update algorithm. In Section 5 simulations based on realistic data from a High-Bit Rate Digital Subscriber Line (HDSL) application illustrate the potential advantages of a filtered-error update. Brief conclusions are offered in Section 6.

2. Problem Definition

The problem is developed using the language of echo cancellation, not because the filtered-error update concept is useful only in that application, but rather to facilitate clarity and precision through a concrete example.

Let $x[n]$ represent a discrete time transmit signal, and $y[n]$ a receive signal. The receive signal is the sum of two components, an echo, $y_0[n]$, which is correlated with samples of $x[n]$, and a "noise" component, $v[n]$, which is uncorrelated with $x[n]$. Contrary to the implication of the label "noise," $v[n]$ is likely a useful signal to some component of the receiver, but from the point of view of the echo canceller it merely degrades performance. The correlation between $x[n]$ and $y_0[n]$ may be defined as a finite convolution sum or as a vector inner product,

$$y_0[n] = \sum_{i=0}^{N-1} x[n-i]h_i = \mathbf{x}^T[n]\mathbf{h}$$

The echo canceller convolves a vector of adaptive coefficients with $x[n]$ to create an echo estimate, $\hat{y}[n] = \mathbf{x}^T[n]\hat{\mathbf{h}}[n]$. This may be termed an *a priori* estimate since it uses coefficient values available at time n. The difference between $\mathbf{h}$ and $\hat{\mathbf{h}}[n]$ is the coefficient error, $\tilde{\mathbf{h}}[n] = \mathbf{h} - \hat{\mathbf{h}}[n]$. The *a priori* prediction error is

$$e[n] = y[n] - \hat{y}[n] = \mathbf{x}^T[n]\tilde{\mathbf{h}}[n] + v[n]$$

The echo estimate and error that are recomputed after the coefficients are updated to $\hat{\mathbf{h}}[n+1]$ are termed *a posteriori*

$$\bar{y}[n] = \mathbf{x}^T[n]\hat{\mathbf{h}}[n+1]$$

$$\in[n] = y[n] - \bar{y}[n] = \mathbf{x}^T[n]\tilde{\mathbf{h}}[n+1] + v[n]$$

Let the coefficients of the filter in Figure 1 be denoted $b_i[n]$. These can be represented compactly as a polynomial in the unit delay operator q^{-1}, $B(n,q^{-1}) = 1 + b_1[n]q^{-1} + \ldots + b_{M-1}[n]q^{-(M-1)}$. The filtered *a priori* error can be written

$$\bar{e}[n] = \sum_{i=0}^{M-1} e[n-i]b_i[n] = B(n,q^{-1})e[n]$$

Similarly, $\overline{\in}[n] = B(n, q^{-1}) \in [n]$. A generalized filtered error substitutes $e[n]$ for $\in [n]$ in $\overline{\in}[n]$, and is thus realizable at time n

$$g[n] = e[n] + \left(B(n, q^{-1}) - 1\right) \in [n]$$

This paper is concerned with an LMS algorithm that uses the generalized filtered error $g[n]$ where $e[n]$ would normally appear in the update

$$\hat{h}[n+1] = \hat{h}[n] + \frac{\mu x[n] g[n]}{1 + \mu x^T[n] x[n]}$$

The operator $B(n, q^{-1})$ is Input Strict Passive (ISP), as defined in [1], if for any input signal $u[n]$, with corresponding output $z[n] = B(n, q^{-1}) u[n]$, the following inequality holds for all integers $L > 0$, for some $\alpha > 0$ and for some finite K,

$$\sum_{n=0}^{L} u[n] z[n] + K \geq \alpha \sum_{n=0}^{L} u^2[n]$$

3. A Deterministic Convergence Theorem:

If $y[n] = 0$, $x[n]$ is bounded, and $B(n, q^{-1})$ is Input Strict Passive (ISP), then

$$\lim_{n \to \infty} e[n] = 0$$

For Proof see appendix.

This theorem is modeled on a deterministic convergence theorem for IIR adaptive algorithms found in [1]. The latter theorem includes other "side" results, which can also be proved here, but have been omitted in the interest of brevity.

The strength of this theorem is its generality, especially in terms of the input signals for which it holds (they need only be bounded). The cost for this strength, however, is that the ISP constraint on $B(n, q^{-1})$ is very difficult to meet for all but the simplest filters. Restating this constraint somewhat loosely, it requires that $B(n, q^{-1})$ not introduce so much phase shift that the output points too far away from the input too often.

In many applications the input signals may be somewhat, or even well constrained. A $B(n, q^{-1})$ that is not ISP may still provide sufficient passivity for these inputs to be perfectly suitable. In particular, a broad band noise $v[n]$ might enhance (stochastic) convergence by introducing energy in $e[n]$ in bands for which $B(n, q^{-1})$ has low phase shift. For a specific implementation, an important convergence metric is the correlation between the actual prediction error $e[n]$ and the filtered error output $\bar{e}[n]$. When the correlation is negative, the update may be divergent as $\bar{e}[n]$ points in the wrong direction.

In some applications, in addition to filtering the echo canceller error, the system also uses nonlinear functions to remove parts of $v[n]$ without disturbing the residual echo. Examples of these are a decision feedback equalizer (DFE) which removes some far end intersymbol interference (ISI), and a decision device (quantizer) which removes the far end transmitted signal. This signal, and the ISI, may be major components of $v[n]$, so removing them may substantially reduce the power of the "noise" that perturbs the echo canceller update. They also create an opportunity for another technique that may enhance convergence of the filtered-error update algorithm, namely the addition of a filter $C\left(n,q^{-1}\right)$, after $B\left(n,q^{-1}\right)$, that approximates the inverse of $B\left(n,q^{-1}\right)$. The combination of the two filters is likely to be closer to ISP. Without the nonlinear functions, this inverse filter would at best recreate $e[n]$, (with a delay), so no advantage could be gained compared to a normal LMS update. With $\bar{e}[n]$ now defined as the output of the combined $B\left(n,q^{-1}\right)$ and $C\left(n,q^{-1}\right)$, the graph in Figure 2 shows how the correlation between $e[n]$ and $\bar{e}[n]$ can be enhanced with only a few taps in the inverse filter.

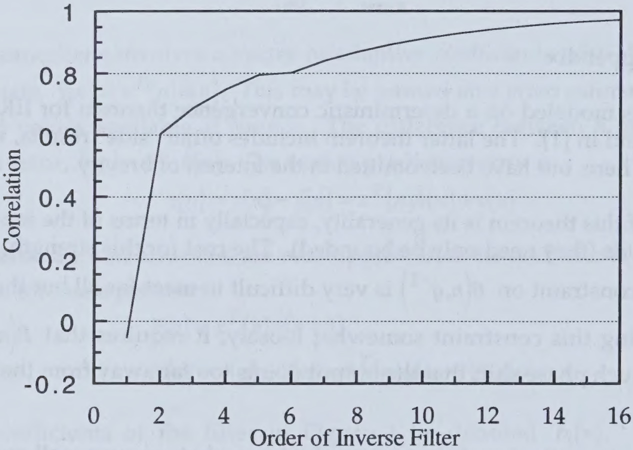


Figure 2: $E[e * \bar{e}]$ vs. tap length of inverse filter $C\left(n,q^{-1}\right)$

In addition to the passivity condition, the theorem adds two other constraints that may be problematic. The first is the requirement that the *a posteriori* error be computed and filtered. This imposes a computational burden that can be onerous. It has been noted in a similar instance, [2], that when the adaptive gain parameter μ

is small, then $\tilde{\boldsymbol{h}}[n+1] \approx \tilde{\boldsymbol{h}}[n]$ and $\in[n] \approx e[n]$. In this case there is practically no difference between using the *a priori* error and the *a posteriori* error, so if convergence is assured in the latter case, it is at least highly likely in the former. Practical experience indicates that the passivity of $B\left(n, q^{-1}\right)$ is of much more significance than whether $e[n]$ or $\in[n]$ is filtered. The data presented in Section 5 of this paper is based on simulations in which only $e[n]$ is computed and filtered.

The second additional constraint is that the "cursor" of the filter $B\left(n, q^{-1}\right)$, be the first coefficient, i.e. that $b_0 = 1$. For many applications this may not be a problem, but in others it will, e.g. if the filter is a linear equalizer that needs to remove precursor ISI. In general, if $b_m = 1$, then the update must use the delayed regressor vector $\boldsymbol{x}[n-m]$. In these cases the conditions of the theorem are not strictly met, but convergence is still likely for most practical cases. For a particular implementation, the metric of interest is the correlation between $\bar{e}[n]$ and $e[n-m]$. When this correlation is positive, convergence should follow. In extreme cases it may be necessary to update $\hat{\boldsymbol{h}}$ only one out of m sample periods.

4. Performance

The previous section examined the convergence behavior of the filtered-error update algorithm. This section assumes (mean square) convergence and quantifies the performance difference between the regular update and the filtered-error update approaches. The stochastic analysis is not as rigorous as that of the previous section, relying on assumptions that are only partly true. The assumptions, like those commonly used in similar analyses [3,4], are justified by their necessity in making the analysis tractable and by the accuracy of the results they produce for realistic situations.

Following are the assumptions that are used in this section.

1. $x[n]$ is a zero mean, white gaussian process with variance σ_x^2.
2. The adaptive gain μ is small enough so that in steady state the elements of $\tilde{\boldsymbol{h}}$ are zero mean and independent of the elements of $\boldsymbol{x}$, and of the noise v.
3. The noise signal $v[n]$ is zero mean, has variance σ_v^2, and is independent of the input process $x[n]$.
4. The term $\mu / \left(1 + \mu \boldsymbol{x}^T[n]\boldsymbol{x}[n]\right)$ is replaced in the update by $\mu / \left(1 + \mu N \sigma_x^2\right)$, which is sometimes abbreviated μ'.

The performance metric of interest is the misadjustment, defined as the ratio of the excess mean squared error (MSE) to the minimum MSE, both measured at the output of the echo canceller [3]. Recalling the equations in section 2, and using the above assumptions,

$$E\left[e^2[n]\right] = E\left[\left(\boldsymbol{x}^T[n]\tilde{\boldsymbol{h}}[n] + v[n]\right)^2\right]$$
$$= \sigma_x^2\, E\left[\tilde{\boldsymbol{h}}^T[n]\tilde{\boldsymbol{h}}[n]\right] + \sigma_v^2$$

It is apparent that the MSE is minimized when $\tilde{\boldsymbol{h}} = 0$, and is therefore equal to σ_v^2. The term $\sigma_x^2\, E\left[\tilde{\boldsymbol{h}}^T\tilde{\boldsymbol{h}}\right]$ thus constitutes the excess MSE, so the misadjustment is

$$\text{Mis} = \frac{\sigma_x^2 E\left[\tilde{\boldsymbol{h}}^T\tilde{\boldsymbol{h}}\right]}{\sigma_v^2}$$

This formula is independent of the method used to update $\tilde{\boldsymbol{h}}$. For regular LMS

$$E\left[\left\|\tilde{\boldsymbol{h}}[n+1]\right\|^2\right] = E\left[\left\|\tilde{\boldsymbol{h}}[n] - \mu'\,\boldsymbol{x}[n]\left(\boldsymbol{x}^T[n]\tilde{\boldsymbol{h}}[n] + v[n]\right)\right\|^2\right]$$
$$= \left(1 - 2\mu'\,\sigma_x^2 + (\mu')^2(N+2)\sigma_x^4\right)E\left[\left\|\tilde{\boldsymbol{h}}[n]\right\|^2\right] + (\mu')^2 N\sigma_x^2\sigma_v^2$$

Therefore

$$\lim_{n\to\infty} E\left[\left\|\tilde{\boldsymbol{h}}^T[n]\tilde{\boldsymbol{h}}[n]\right\|\right] = \frac{\mu N\sigma_v^2}{2\left(1+\mu N\sigma_x^2\right) - \mu(N+2)\sigma_x^2} = \frac{\mu N\sigma_v^2}{2+\mu(N-2)\sigma_x^2}$$

The misadjustment for the regular LMS update (using $e[n]$) is thus

$$\text{Mis}_e = \frac{\mu N\sigma_x^2}{2+\mu(N-2)\sigma_x^2}$$

For the filtered-error update, things get more messy. To the above assumptions add the following approximation:

$$E\left[\boldsymbol{x}[n-i]\boldsymbol{x}^T[n]\boldsymbol{x}[n]\boldsymbol{x}^T[n-i]\right] = (N+2)\sigma_x^4 \cdot I \text{ for } 1 \le i \le M-1.$$

The true expectation is a diagonal matrix with $(N+2)\sigma_x^4$ for the first *N* - *i* entries and $N\sigma_x^4$ for the last *i* entries, so for reasonably large values of *N* this approximation is very close. Recall also that since $v[n]$ is zero mean so is $\bar{v}[n]$. The filtered-error update is

$$\begin{aligned}\tilde{\boldsymbol{h}}[n+1] &= \tilde{\boldsymbol{h}}[n] - \mu' \boldsymbol{x}[n]g[n] \\ &= \tilde{\boldsymbol{h}}[n] - \mu' \boldsymbol{x}[n]\left[e[n] + \left(B\left(n,q^{-1}\right)-1\right)\in[n]\right] \\ &= \left(I - \mu' \boldsymbol{x}[n]\boldsymbol{x}^T[n]\right)\tilde{\boldsymbol{h}}[n] - \mu' \boldsymbol{x}[n]\bar{v}[n] \\ &\quad - \mu' \boldsymbol{x}[n]\sum_{i=1}^{M-1} b_i \boldsymbol{x}^T[n-i]\tilde{\boldsymbol{h}}[n-i+1]\end{aligned}$$

Taking the norm square of both sides and then taking expectations, all the cross terms on the right side are zero because of the assumptions. What remains is

$$\begin{aligned}E\left[\left\|\tilde{\boldsymbol{h}}(n+1)\right\|^2\right] &= \left(1 - 2\mu'\sigma_x^2 + (\mu')^2(N+2)\sigma_x^4\right)E\left[\left\|\tilde{\boldsymbol{h}}(n)\right\|^2\right] \\ &\quad +(\mu')^2 N\sigma_x^2\sigma_v^2 + (\mu')^2(N+2)\sigma_x^4\sum_{i=1}^{M-1} b_i^2 E\left[\left\|\tilde{\boldsymbol{h}}(n-i+1)\right\|^2\right]\end{aligned}$$

In steady state we assume $E\left[\left\|\tilde{\boldsymbol{h}}(n+i)\right\|^2\right] = E\left[\left\|\tilde{\boldsymbol{h}}(n)\right\|^2\right] = E\left[\left\|\tilde{\boldsymbol{h}}^T\tilde{\boldsymbol{h}}\right\|^2\right]$. Bringing all these terms to the left and then dividing through yields

$$\begin{aligned}\lim_{n\to\infty} E\left[\left\|\tilde{\boldsymbol{h}}^T[n]\tilde{\boldsymbol{h}}[n]\right\|^2\right] &= \frac{(\mu')^2 N\sigma_x^2\sigma_{\bar{v}}^2}{2\mu'\sigma_x^2 - (\mu')^2(N+2)\sigma_x^4\sum\limits_{i=0}^{M-1} b_i^2} = \frac{\mu N\sigma_{\bar{v}}^2}{2\left(1+\mu N\sigma_x^2\right) - \mu(N+2)\sigma_x^2\sum\limits_{i=0}^{M-1} b_i^2} \\ &= \frac{\mu N\sigma_{\bar{v}}^2}{2 + \mu\sigma_x^2\left(N - 2 - \sum\limits_{i=1}^{M-1} b_i^2\right)}\end{aligned}$$

Therefore the misadjustment for the filtered-error update (using $g[n]$) is

$$Mis_g = \frac{\mu N\sigma_{\bar{v}}^2\sigma_x^2}{\left[2 + \mu\sigma_x^2\left(N - 2 - \sum\limits_{i=1}^{M-1} b_i^2\right)\right]\sigma_v^2}$$

In the usual case where $\sum\limits_{i=1}^{M-1} b_i^2 << N - 2$, Mis_g is approximated by

$$Mis_g = \frac{\mu N\sigma_{\bar{v}}^2\sigma_x^2}{\left[2 + \mu\sigma_x^2(N-2)\right]\sigma_v^2} = Mis_e \cdot \frac{\sigma_{\bar{v}}^2}{\sigma_v^2}$$

Therefore, the misadjustment improvement for the filtered-error update should be the same as the attenuation experienced by the noise when it passes through the filter. Should there also be non-linear elements present in the system (e.g. DFE) that subtract parts of $v[n]$, these losses are reflected in the attenuation factor, $\sigma_{\bar{v}}^2/\sigma_v^2$, as well.

5. Realistic Implementation: HDSL

To verify the Misadjustment improvement predicted in the previous section it is useful to apply the filtered-error update in a realistic implementation. This section presents simulation results, based on such an implementation, that demonstrate the performance advantage. The simulation uses realistic data from a High bit-rate Digital Subscriber Line (HDSL) system. An HDSL transceiver employs full duplex communication over ordinary twisted-pair copper wires at 784 kbits/sec. At distances up to 12,000 feet it must maintain a bit error rate of less than 10^{-7}. The "noise" part of the receive signal consists principally of the far end signal (including ISI), and near end cross talk (NEXT). Figure 3 shows the main transceiver signal processing blocks: echo canceller, linear equalizer, decision feedback equalizer and quantizer.

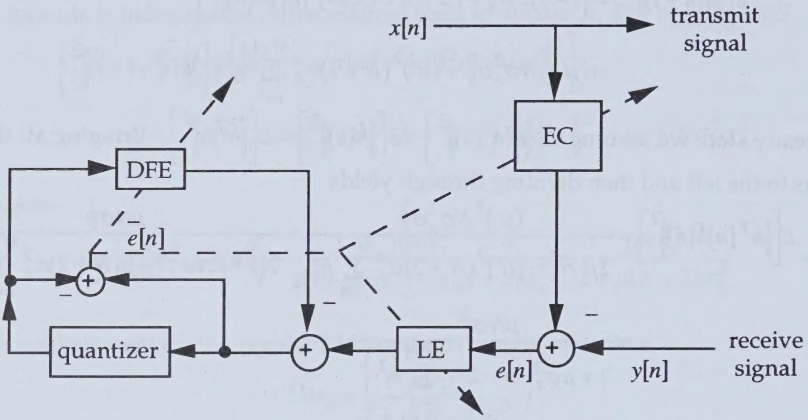


Figure 3: HDSL block diagram

The final quantizer error consists mainly of the *a priori* residual echo and the NEXT, both filtered by the LE. No *a posteriori* estimates are used in the simulation. It is this signal, $\bar{e}[n]$ in the Figure, that is used in the filtered-error update simulation. The regular LMS simulation uses $e[n]$. Figure 4 shows the Misadjustment (i.e. residual echo power normalized by σ_v^2) vs. time for both updates. The simulation uses realistic echo path (**h**) and LE coefficients and realistic signal and NEXT levels as well. The LE and DFE are initialized to their optimal values and held constant throughout the simulations. The adaptive gain parameter is $\mu = 10^{-5}$, $N = 128$ and $\sigma_x^2 = 5$. In this simulation the ratio $\sigma_{\bar{v}}^2 / \sigma_v^2 = -35$ dB, which is reflected in the steady state Misadjustment difference. This data lends credibility to the MSE analysis in the previous section, and demonstrates clearly the potential advantage of a filtered-error update.

6. Conclusions

This paper has proposed a filtered-error update as an alternative to normal LMS. The advantage of this approach is that the Misadjustment may be reduced by a factor equal to the amount of attenuation applied to the noise as it is filtered. A simulation based on realistic HDSL data demonstrates this advantage. Of equal importance is that the performance advantage may be achieved with no increase in system complexity because the filtered error is already present. The paper also presents a deterministic convergence theorem, with proof. Though the theorem imposes a difficult constraint on the filter, practical experience indicates that convergence is readily achieved even if the constraint is not met.

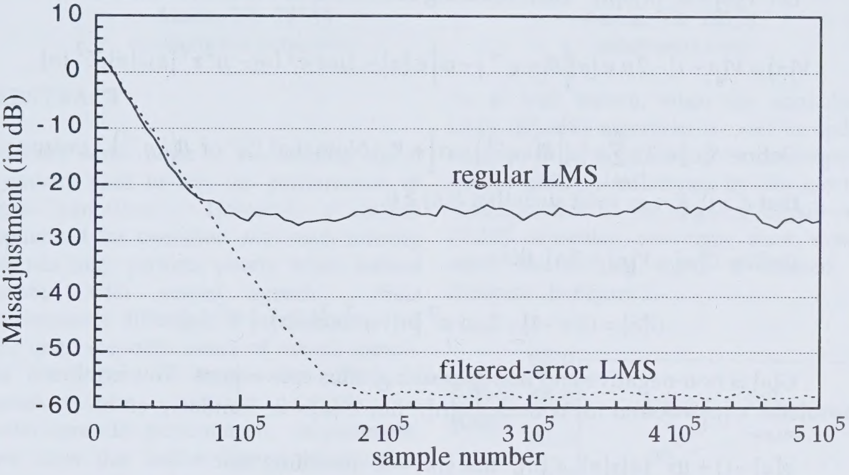


Figure 4: Misadjustment vs. time for regular and filtered-error updates

Appendix

Proof of Convergence Theorem:

Multiplying the update equation by $x^T[n]$, and subtracting both sides from $y[n]$ yields

$$\in[n] = e[n] - \frac{\mu x^T[n]x[n]g[n]}{1+\mu x^T[n]x[n]}$$

Adding $\left[B\left(n,q^{-1}\right)-1\right]\in[n]$ to both sides produces

$$\overline{\in}[n] = \frac{1}{1+\mu x^T[n]x[n]}\, g[n]$$

Substituting into the coefficient error update and rearranging yields

$$\tilde{\boldsymbol{h}}[n+1] + \mu \boldsymbol{x}^T[n]\,\overline{\in}[n] = \tilde{\boldsymbol{h}}[n]$$

Let $V[n] = \tilde{\boldsymbol{h}}^T[n]\tilde{\boldsymbol{h}}[n]$. Then squaring both sides of the above leads to

$$V[n] = V[n-1] - 2\mu \in[n]\left[B\left(n, q^{-1}\right) - \alpha\right] \in[n] - 2\mu\alpha \in^2[n] - \mu^2 \boldsymbol{x}^T[n]\boldsymbol{x}[n]\,\overline{\in}^2[n]$$

Define $S[n] = 2\mu \sum_{i=1}^{n} \in[i]\left[B\left(i, q^{-1}\right) - \alpha\right] + K$. Note that ISP of $B\left(i, q^{-1}\right)$ ensures that a > 0, $K < \infty$ exist such that $S[n] \geq 0$.

Define $G[n] = V[n] + S[n]$, then

$$G[n] = G[n-1] - 2\mu\alpha \in^2[n] - \mu^2 \boldsymbol{x}^T[n]\boldsymbol{x}[n]\,\overline{\in}^2[n]$$

$G[n]$ is non-negative and non-increasing, thus convergent. This implies $\lim_{n\to\infty} \in[n] = 0$, and for bounded x[n], $\lim_{n\to\infty} \overline{\in}[n] = 0$. Similarly, since $g[n] = \left(1 + \mu x^T[n]x[n]\right)\overline{\in}[n]$, $\lim_{n\to\infty} g[n] = 0$. Recalling that $e[n] = g[n] - \overline{\in}[n] + \in[n]$, since each of the elements on the right side converges to zero

$$\lim_{n\to\infty} e[n] = 0$$

This concludes the proof.

References

1. G. C. Goodwin and K. S. Sin, Adaptive *Filtering, Prediction and Control.* Englewood Cliffs, NJ: Prentice-Hall, 1984.
2. M. G. Larimore, J. R. Treichler and C. R. Johnson, Jr., "SHARF: an algorithm for adaptive IIR digital filters," *IEEE Trans. Acoust. Speech Signal Process.*, vol. ASSP-28, no. 4, pp. 428-440, Aug. 1980.
3. B. Widrow *et al.*, "Stationary and nonstationary learning characteristics of the LMS adaptive filter," *Proc. IEEE*, vol. 64, no. 8, pp. 1151-1162, Aug. 1976.
4. S. Haykin, *Adaptive Filter Theory*, Englewood Cliffs, NJ: Prentice-Hall, 1986.

EFFECTS OF QUASI–PERIODIC TRAINING SIGNALS ON ACOUSTIC ECHO–CANCELLER PERFORMANCE

Shawn McCaslin

Crystal Semiconductor Corporation
4210 South Industrial Drive
Austin, TX 78744
srm@crystal.cirrus.com

Nicholas van Bavel

Crystal Semiconductor Corporation
Now at: Oasis Design Incorporated
4208 Avenue H
Austin, TX 78751
nvb@oasis.com

ABSTRACT

Filtered white noise is the training signal typically used to test the performance of echo cancellers. However, a system optimized for operation with such training signals may perform poorly when trained instead with voiced speech. This performance difference is a consequence of the quasi–periodic nature of voiced speech. In this paper, we explore the effects of quasi–periodic training signals on echo–canceller performance. In particular, we show that unlike filtered white noise, quasi–periodic training signals can actually limit the asymptotic echo–return loss of echo cancellers. In addition, we examine the ability of several existing techniques to improve performance when quasi–periodic signals are used to train echo cancellers.

INTRODUCTION

The basic architecture for adaptive–FIR echo cancellers is shown in Figure 1.

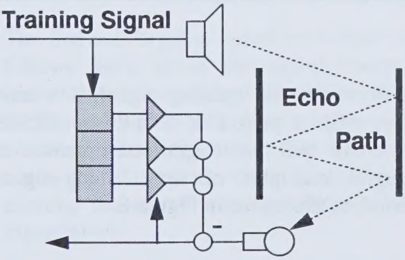


Figure 1 --Acoustic Echo Cancellation

As is well known, when the normalized LMS (NLMS) algorithm is used to update adaptive–filter coefficients, the convergence rate is strongly influenced by the spectral characteristics of the training signal––the NLMS algorithm converges more slowly when the training signal is colored, as illustrated in Figure 2.

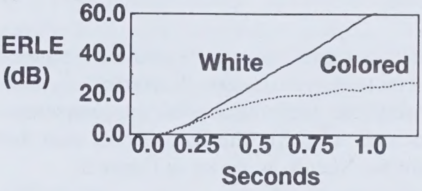


Figure 2--NLMS with White & Colored Training Signals

One technique that can be used to compensate for spectral coloring is frequency–domain adaptive filtering, which is illustrated in Figure 3 [1,2].

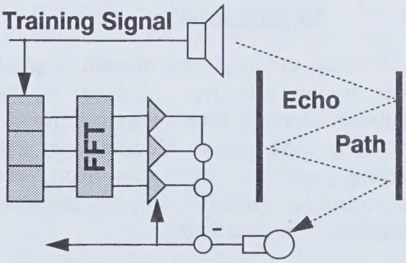


Figure 3 --Frequency-Domain EC

In frequency–domain adaptive filtering, an FFT is used to decorrelate the training signal. This technique is very effective in particular with filtered white noise, as shown in Figure 4; frequency–domain adaptive filtering produces nearly the same performance with a colored training signal as the NLMS produces with a white training signal.

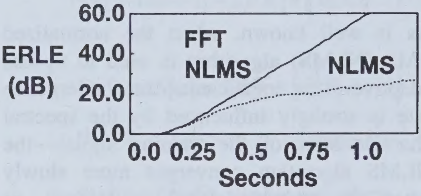


Figure 4--FFT NLMS & NLMS: Filtered White-Noise Training Signal

If, however, a quasi–periodic, power–stationary, voiced–speech training signal is used, the performance of the frequency–domain adaptive filter is no better than that for the NLMS, as shown in Figure 5.

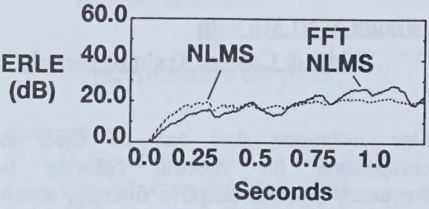


Figure 5--FFT NLMS & NLMS: Continuous Voiced-Speech Training Signal

In this case, the frequency–domain adaptive filter is not effective. Indeed, if the frequency–domain filter is trained for an arbitrarily long period of time with this training signal, the echo–return–loss enhancement (ERLE) never increases significantly beyond 30 dB. The main contribution of this paper is to explain this limit on the performance of this frequency–domain adaptive filter when trained with quasi–periodic signals, and, more generally, to explore the effects of quasi–periodic training signals on echo–canceller performance.

EFFECTS OF PERIODIC TRAINING SIGNALS ON ECHO CANCELLERS

Consider the diagram shown in Figure 6.

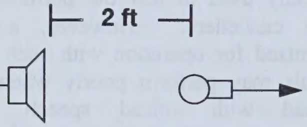


Figure 6 -- Speaker & Mic in Free Space

Here, a speaker and a microphone are placed two feet apart in free space. Since sound propagates in air at about 1 ft/ms, if the sampling rate is 8 kHz, the true echo–path response in this case would consist of an impulse at index 16 (2 ft x .001 s/ft x 8000 samples/s = 16) in the echo–canceller filter, as illustrated in Figure 7.

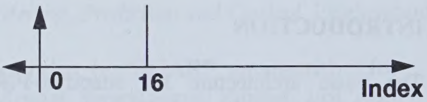


Figure 7 -- Ideal Free-Space Path Estimate

If, however, the training signal is a sine wave with a period of 2 ms, the NLMS algorithm will create spurious impulses at integral multiples of the training–signal period, as illustrated in Figure 8:

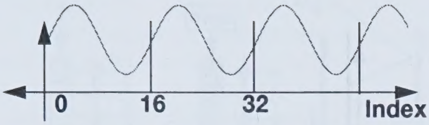


Figure 8 --Free-Space Path Estimate with a Sine-Wave Training Signal

The signal at the microphone looks the same at delays equal to integral multiples of the period of the training signal (i.e., the autocorrelation function of a sine wave is periodic). As a result, the adaptive filter can set the coefficient at each of those delays to any value, as long as the sum of these coefficients is the value required to cancel the sine wave. Even though the resulting estimated path response is very different from the true (i.e., white–noise trained) path response, the canceller will continue to cancel the echo well, as long as the training–signal remains a sine wave at the given frequency.

In addition, the echo canceller will converge rapidly, more rapidly than it does for white noise. This contradicts the widely–held view that convergence speed is always limited by the size of the smallest eigenvalue of the training signal, and that convergence is always fastest for white noise. For a sine wave, the smallest eigenvalue is zero (indeed, all eigenvalues of a sine wave but one are zero), implying an infinitely slow convergence rate.

The fast convergence speed for a sine wave follows since all of the signal energy is concentrated in one large eigenvalue, which consequently gives a large convergence time constant. Zero eigenvalues do not matter since, by definition, there is no energy exciting the system corresponding to those eigenvalues.

If the sine wave for Figure 6 were left on, but reflecting walls were placed around the microphone and speaker, the reflections would still add at the microphone to produce a sine wave, though the amplitude and phase may have changed. As a result, the position and amplitude of the impulses in the estimated path response may shift, but they would all still be equally spaced, independent of the path response. Clearly, the true (white noise) path response cannot be determined when a sine wave is used as the training signal.

Worse yet, the echo canceller's performance in this case is exquisitely sensitive to the path response. An added reflection that causes a 2% change in phase will cause the entire path estimate to shift by one index. Coefficients that were zero must reconverge to a large nonzero value, and large–valued coefficients must reconverge to zero. The canceller would converge faster and suffer a smaller performance degradation if it were to start from all zero coefficients instead!

EFFECTS OF QUASI–PERIODIC TRAINING SIGNALS ON ECHO CANCELLERS

Similarly, small changes in the frequency of the training signal can also require radical reconvergence. For example, if the training–signal period in the above example decreases by only 2%, the impulse at index 64 will shift to index 63. In this case, a coefficient that was zero (at index 63) will have to readapt to a large nonzero value; the nonzero tap at 64 will have to reconverge to zero.

This phenomena has a strong influence on the performance of echo/noise cancellers when they are trained with quasi–periocic signals like voiced speech or car engine noise. These signals are quasi–periodic, since they look periodic over short time intervals, but the period of their fundamental changes versus time. When echo cancellers are trained with such signals, spurious impulses form at higher indices while the fundamental is stable, but must move as the frequency changes. If the adaptive filter cannot adapt quickly enough to compensate, the echo-canceller's performance will suffer.

This effect is illustrated in the following examples, where the true path response is that shown in Figure 7. In the first example (Figure 9), the training signal is generated by extracting a single period of data from real speech and repeating it. In this case, the data is strictly periodic, but its spectrum is otherwise speechlike.

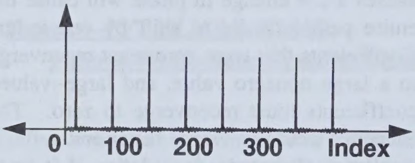


Figure 9 --Free-Space Path Estimate Periodic Training Signal

With this training signal, spurious impulses appear periodically in the response estimate. As before, the canceller will perform well as long as the period of the training signal does not change. If, however, the signal is made quasi-periodic, echo-cancellation performance will be reduced, as illustrated in Figure 10. Here, power-stationary, real speech was used as the training signal with the NLMS algorithm. In this case, spurious impulses are significantly suppressed, but now they significantly interfere with canceller performance. As a result of the motion of the spurious impulses, echo cancellation performance is limited to only about 20 dB!

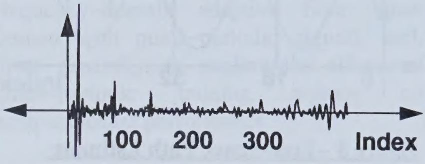


Figure 10--Free-Space Path Estimate, Quasi-periodic Training Signal

MITIGATING THE EFFECTS OF QUASI-PERIODIC TRAINING SIGNALS

Unfortunately, echo cancellers must be trained with quasi-periodic signals in many practical applications. Techniques that reduce the performance degradation caused by training with quasi-periodic signals would have great utility.

We have identified two such techniques: pre-emphasis [3] and graded update gain [4]. Neither technique is new; our contribution here is recognizing how they can be adapted to address the problem of quasi-periodic training signals.

PRE-EMPHASIS

Classically, pre-emphasis, as illustrated in Figure 11, is used to whiten colored training signals. By whitening the training signal, a pre-emphasis filter ("emp" in Figure 11) can permit an NLMS filter to approach its white-noise convergence speed.

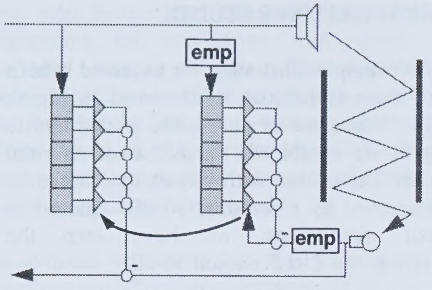


Figure 11 -- Pre-emphasis

Pre-emphasis can also be used to reduce the effects of quasi-periodic training signals, thereby improving the asymptotic ERL of adaptive filters. For quasi-periodic training signals, the pre-emphasis filter can be chosen to remove (or greatly attenuate) the fundamental of the quasi-periodic component. This significantly reduces the tendency for spurious impulses to form. For quasi-periodic training signals like voiced speech, the higher harmonics shift about in frequency quickly enough as to not allow spurious impulses to interfere.

We have also experimented with adaptive pre-emphasis filters [5], and found that they produce similar performance benefits. Unfortunately, they remove too much of the training signal, causing the ERLE of the echo canceller to suffer significantly when the near-end environment is noisy. That observation, along with the greater complexity of adaptive pre-emphasis filters, leads us to favor fixed pre-emphasis filters.

GRADED UPDATE GAIN

Another effective technique is to grade the update gain versus index. To understand the operation of the graded update-gain NLMS adaptive filter, consider the echo-path response shown in Figure 12.

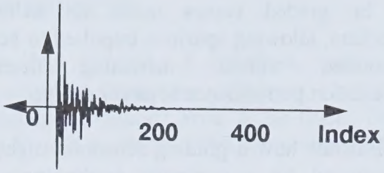


Figure 12 -- Real Car Echo-Path Response

This figure shows an echo-path response that was measured in a car. As one might expect, there is a short period at the beginning of the where the path response is zero. This corresponds to the time required for sound to propagate from the speaker to the microphone.

The path response that immediately follows this delay is the earliest arriving (probably directly from the speaker), and therefore, the strongest. The echo that arrives later travels further, diffuses, and reflects off of acoustically-absorbent materials. It is, therefore, weaker.

This decay characteristic should follow in most car-interior echo paths. Although the actual values of the path-response coefficients may vary (particularly in sign), the general shape (like that in Figure 12) should always be the same.

The graded update-gain NLMS adaptive filter takes advantage of this characteristic. A relatively large update gain is used for large-valued, low-index (early) coefficients. Smaller update gain is used to update smaller-valued, high-index (later) coefficients. Canceller coefficients for weak, late-arriving echo coincide with the spurious impulses that are produced by quasi-periodic speech. Smaller gains on these coefficients suppress spurious impulses. In summary, since the echo response typically decays versus time for most real-world echo paths, the update gain can be graded versus index for echo cancellers, allowing spurious impulses to be suppressed without detracting from cancellation performance at larger delays.

To illustrate how a grading schedule might be selected for a particular environment, again consider the car environment. Our experiments suggest that echo strength in a car decays at about 1 dB per ms. A reasonable grading schedule in this case would be to reduce the update gain by 2x (6 dB) every 64 taps (8 ms), as illustrated in Figure 13 [4]. This grading schedule provides a substantial improvement in echo suppression for the type of echo response shown in Figure 12.

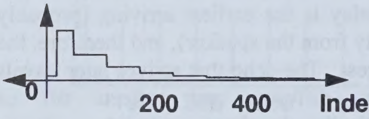


Figure 13 -- Beta Grading Schedule

This grading schedule also provides a low update gain at the 16 earliest taps, which assumes that the microphone will never be less than 2 ft from the speaker. Such a minimum-distance restriction should be reasonable, since the acoustic coupling between the microphone and the speaker would probably need to be limited in practice. (If the microphone and the speaker are too close, the echo canceller may be unable to adequately attenuate the echo). Of course, the grading schedule in Figure 9 will not produce the best-possible results in all environments, since it was selected for the car echo-path response shown in Figure 8. For example, a conference room would probably require a shallower decay profile.

SIMULATION RESULTS

An example illustrating the expected benefit of these techniques is illustrated in Figure 14. When pre-emphasis and graded update gain are combined, the ERLE of an echo canceller trained with voiced speech can be improved by more than 10 dB. Indeed in our simulations, we have seen the asymptotic ERLE exceed 50 dB.

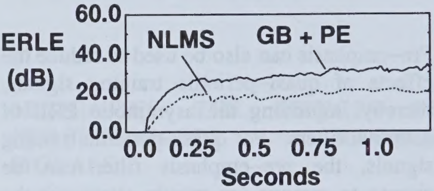


Figure 14: NLMS vs. NLMS w/ Pre-Emphasis + Graded Update Gain; Continuous Voiced-Speech Training Signal

CONCLUSIONS/RECOMMENDATIONS

In this paper, we have outlined some of the effects that quasi-periodic training signals have on echo-canceller performance––in particular, we have shown that unlike filtered white noise, quasi-periodic training signals can limit the asymptotic ERLE of echo/noise cancellers. We have also identified some existing techniques that help loosen the performance limit imposed by quasi-periodic training signals.

We believe these results should influence practical aspects of echo-canceller development and testing. In particular, we recommend that echo-canceller standards for speech applications (like CCITT G.165) be modified to specify required ERLE performance with quasi-periodic training signals.

We also recommend that when creating algorithms for application with quasi–periodic training signals (like speech) echo–canceller algorithm researchers report the asymptotic ERLE of their algorithms. They should also report convergence speed when their algorithms are trained with quasi–periodic training signals.

In addition, we request that other researchers continue our efforts to understand the effects of signal statistics on system–identification estimates, not just on convergence speed. For example, a few questions that need to be answered are:

Do adaptive filter algorithms optimized for performance with auto–regressive training signals help mitigate the effects of quasi–periodic training signals?

What is the performance of RLS adaptive filters with quasi–periodic training signals?

We have observed that the frequency–domain adaptive filter shown in Figure 3 does little to mitigate the effects of quasi–periodic training signals. We have observed, however, that frequency–domain adaptive filters in which independent error signals are used for each frequency band appear to work well with quasi–periodic training signals [6]––why?

REFERENCES

[7] M. Dentino, J. McCool, and B. Woodrow, "Adaptive Filtering in the Frequency Domain", Proc. of the IEEE, Vol. 66, No. 12, pp. 1658–1659, December 1978.

[2] S. S. Narayan and A. M. Peterson, "Frequency Domain Least–Mean–Square Algorithm", Proc. of the IEEE, Vol. 69, No. 1, pp. 124–126, January 1981.

[3] U. Schultheiβ, "Uber die Adaption eines Kompensators fur akustische Echos", Fortschr–Ber. VDI Teihe 10, Nr. 90, VDI–Verlag Dusseldorf, 1988.

[4] S. Makino, Y. Kaneda, N. Koizumi, "Exponentially Weighted Stepsize NLMS Adaptive Filter Based on Statistics of a Room Impulse Response", IEEE Trans. on Speech and Audio Processing, Vol. 1, No. 1, January 1993

[5] S. Yamamoto, S. Kitayama, J. Tamura, and H. Ishigami, "An Adaptive Echo Canceller with Linear Predictor", Trans. of the IECE of Japan, E62(1979), pp. 851–857.

[6] F. Dohnal, "Generalized Frequency Domain Adaptive Filter and its Application for Acoustic Echo Canceller", Proc. of the 3rd International Workshop on Acoustic Echo Control, Plestin les Greves, France, September 1993

Model Set Selection for Complexity Reduction in Adaptive Filters

Sergio Zimmermann* and Geoffrey A. Williamson†
Department of Electrical and Computer Engineering
Illinois Institute of Technology
Chicago IL 60616

Abstract

The problem of selecting a model set for adaptive filters is considered, with the goal of using *a priori* knowledge about the filter operating environment to reduce the complexity of the implementation. A number of different methods are presented to achieve this objective when the *a priori* information comes in the form of a collection of example optimal filters. An adaptive filter structure having fixed, but nonzero, pole locations is considered; the methods for model set selection are tantamount to a variety of approaches for choosing the fixed poles. The methods are compared in terms of the mean square error performance of the adaptive filter as a function of the number of adapted parameters. It is shown that the fixed pole adaptive filters can achieve a performance level comparable to that for standard FIR filters while requiring fewer adapted parameters, and hence less filter complexity.

1 Introduction

Finite impulse response (FIR) adaptive filters have proven to be the most widespread identification structure in adaptive signal processing applications [5, 19]. This popularity stems in part from the globally convergent parameter behavior of adaptive algorithms for these structures [10]. However, when the ideal filter response is very long, they have important disadvantages in relation to pole/zero adaptive filters. First, the large number of adapted parameters necessary for modeling long impulse responses imposes an excessive computational burden. Second, the misadjustment caused by residual parameter motion at convergence can be large if many parameters are adjusted. Pole/zero adaptive filters have been investigated as possible alternatives [6], since they require fewer adjustable parameters for modeling long impulse responses. However, these models have not gained widespread acceptance because they may achieve only suboptimal performance due to multimodal error surfaces [14], or because they require satisfaction of a strict positive reality condition [7].

In this paper we consider a new class of adaptive filter, termed a Fixed Pole Adaptive Filter (FPAF). The FPAF preserves the global convergence of FIR adaptive filters, while enhancing the ability to model long impulse responses with a small number of adapted

---

*Partially supported by CNPq, Brazil.
†Partially supported by the National Science Foundation under Grant MIP-9102620.

parameters. Adaptive filters in this class possess both zeros and poles, and thus have an infinite impulse response. However, the poles are constrained to fixed locations, chosen in order to reflect the available *a priori* information about the desired filter response. This information can arise in different forms, depending on the most suitable way to gather it. We present herein several systematic methods for selecting the pole locations which determine the model set. These methods are based on having available a collection of example impulse or input-output responses which typify the range of desired responses. The advantage achieved by the FPAF is a reduced mean square error (MSE) for a given number of adapted parameters.

Related structures have recently been considered in system identification and adaptive filtering. Model classes derived from Legendre polynomials [2, 11], Laguerre polynomials [2, 3, 17], as well as those using Kautz filters [18] can be represented as special cases of the FPAF. Also fitting into the FPAF framework are the "Gamma filters" of [12] and standard FIR models. Additionally, the FPAF lies within the paradigm of the Vector Space Adaptive Filter [20, 21]. Kaelin *et al.* also considers the use of fixed poles in system modeling [8], with the poles based on information provided by a number of example systems.

The paper is organized in the following way: Section 2 describes the FPAF. Section 3 shows different methods for selecting the fixed pole locations, and Section 4 describes two computer experiments which illustrate the advantages of the FPAF approach. The experiments show that the FPAF can achieve improved performance in relation to FIR and Kautz based adaptive filters.

2 Fixed Pole Adaptive Filters

The Fixed Pole Adaptive Filter, which we have introduced in [22], is indicated in Fig. 1. For simplicity, we have shown a filter structure having only real poles; it is straightforward to generalize the structure to admit complex conjugate pole pairs, which will arise for the majority of applications. The parameters $\{a_i\}$ in Fig. 1 are the *fixed* pole locations of the filter, and the parameters $\{b_i(k)\}$ are the adapted coefficients which weight the intermediate signals $\{x_i(k)\}$ to generate the output. It is possible to show that by choices of the $\{b_i\}$ coefficients, any transfer function of the form

$$H(z) = \frac{c_0 z^n + c_1 z^{n-1} + \cdots + c_n}{(z-a_1)(z-a_2)\cdots(z-a_n)}, \qquad c_i \in \Re \tag{1}$$

can be realized by the adaptive filter. Therefore, one sees that this filter structure can adapt to any nth order transfer function that has poles constrained to locations selected by the adaptive filter designer.

The motivation of the FPAF structure is as follows. If the filter's poles are chosen reasonably close to the dominant poles of the ideal filter, then a small mean square error can be achieved for a relatively small value of n, the number of adapted parameters.

3 Pole Selection

To select the fixed pole locations in an appropriate fashion, it is necessary to have *a priori* information about the set of ideal filters that the FPAF will model. We present here several systematic methods for pole selection, based on availability of a set of example

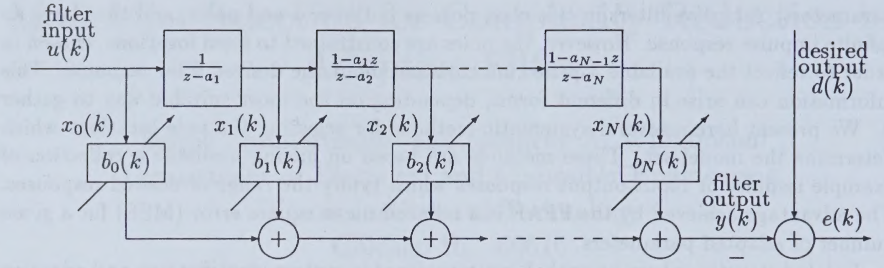


Figure 1: Configuration of the adaptive filter.

filters which reflect this range of expected ideal filter responses. The way that these examples are generated may vary, depending on the specific application.

Figure 2 shows a diagram depicting different steps and starting points to evaluate the pole locations. We consider three starting points: availability of M impulse responses $\{h_\ell(k)\}_{\ell=1}^{M}$, M input-output responses, or M state space descriptions $\{A_\ell, b_\ell, c_\ell, d_\ell\}_{\ell=1}^{M}$ of the example filters. There are many different procedures that the FPAF designer can use to flow through the diagram; however, we will present only those which we consider most appropriate and those which we have tested. Appearing below are descriptions of each of these procedures, labelled by the letters (A) – (F) which are given next to each branch of Fig. 2. (A) and (B) represent new techniques which we have developed for FPAF pole selection. The other branches in the figure represent either our earlier work, or the work of other researchers that may be brought to bear on some aspect of this problem. In particular we note the work of [8], represented by (C), which can be used to find FPAF pole locations based on example impulse responses.

A. Least Squares

It is possible to select poles for the FPAF from example desired input-output responses by using the method of least squares. We assume that we have available M input-output responses

$$\mathbf{y}_\ell = [y_\ell(k-1)\ y_\ell(k-2)\ \cdots y_\ell(0)]^T, \quad \ell = 1, \ldots, M,$$

resulting from the input sequence

$$\mathbf{u} = [u(k-1)\ u(k-2)\ \cdots u(0)]^T,$$

with k the number of samples. We model the input-output relationship by

$$y_\ell(n) = \sum_{i=1}^{M} \alpha_{\ell i} y_\ell(n-i) + \sum_{j=1}^{Z} \beta_{\ell j} u(n-j),$$

where M is the number of poles and Z the number of zeros. The LS estimation minimizes

$$J = \sum_{\ell=0}^{M} \sum_{n=M}^{k} e_\ell^2(n) = \sum_{\ell=0}^{M} \sum_{n=M}^{k} \left(y_\ell(n) - \hat{y}_\ell(n)\right)^2$$

with

$$\hat{y}_\ell(n) = \sum_{i=1}^{M} \hat{\alpha}_i y_\ell(n-i) + \sum_{j=1}^{Z} \hat{\beta}_j u(n-j)$$

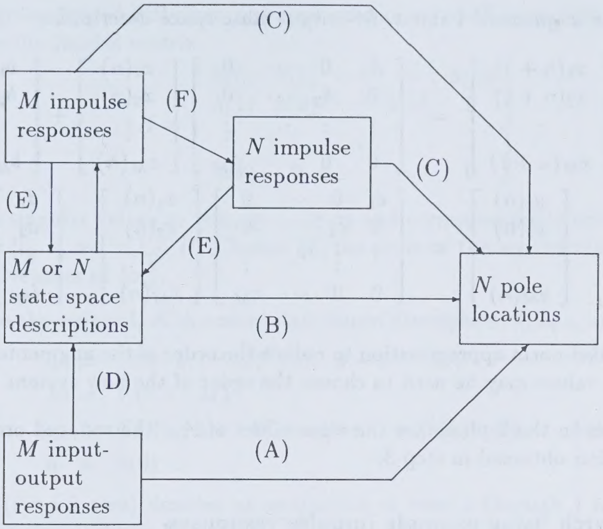


Figure 2: Routes to selection of pole locations.

by choice of $\{\hat{\alpha}_i\}$ and $\{\hat{\beta}_i\}$. In essence, we are fitting *one* system simultaneously to M different systems in a least square sense. It is straightforward to extend the standard least squares solution to this problem. The pole locations $\{a_i\}$ for the FPAF are then given by the roots of the polynomial

$$x^M - \hat{\alpha}_1 x^{M-1} - \hat{\alpha}_2 x^{M-2} - \cdots - \hat{\alpha}_M = 0.$$

The advantages of this method are low computational cost and good immunity to white noise added to the measured output of the system. This LS method is very useful as a quick test of the FPAF approach.

B. Transformation of M state space descriptions into N pole locations

Given a number of example ideal filters in state-space form, it is also possible to specify a set of pole locations for the FPAF. The method we give here is based in the Hankel-Norm approximation [4] of linear systems. We are given a set of example systems formed by state space descriptions. A 1-input, M-output system is formed, with M the number of example systems, and Hankel-norm techniques used to reduce the order of this system. However, frequently one does not originally have the example systems in state space form, which requires an intermediate step to find this description for each example system (see techniques (D) and (E)). The method proceeds as follows:

1. Assume that we have available the 1-input, 1-output state space descriptions A_ℓ, b_ℓ, c_ℓ and d_ℓ of each example system for $\ell = 1, \ldots, M$.

2. Form the augmented 1-input, M-output state space description

$$\begin{bmatrix} x_1(n+1) \\ x_2(n+1) \\ \vdots \\ x_M(n+1) \end{bmatrix} = \begin{bmatrix} A_1 & 0 & \cdots & 0 \\ 0 & A_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & A_M \end{bmatrix} \begin{bmatrix} x_1(n) \\ x_2(n) \\ \vdots \\ x_M(n) \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_M \end{bmatrix} u(n)$$

$$\begin{bmatrix} y_1(n) \\ y_2(n) \\ \vdots \\ y_M(n) \end{bmatrix} = \begin{bmatrix} c_1 & 0 & \cdots & 0 \\ 0 & c_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & c_M \end{bmatrix} \begin{bmatrix} x_1(n) \\ x_2(n) \\ \vdots \\ x_M(n) \end{bmatrix} + \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_M \end{bmatrix} u(n)$$

3. Use Hankel-norm approximation to reduce the order of the augmented system. The singular values may be used to choose the order of the new system.

4. The poles in the z-plane are the eigenvalues of A_R, the reduced order state space description obtained in step 3.

C. Pole search using example impulse responses

If we begin with a set of impulse response descriptions for the example ideal filters, we may find pole locations for the FPAF using the iterative method given in [8]. This method, when applied in the single system case ($M = N = 1$), corresponds to the Steiglitz-McBride method mentioned in D. below. Let $\{h_i\}$ be the collection of example impulse responses. The method of [8] is designed to minimize by choice of poles $p_1, \ldots, p_N$ the cost function

$$\sum_{i=1}^{M} \rho_i^2 \left\| h_i - \hat{h}_i(p_1, \ldots, p_N) \right\|^2 . \tag{2}$$

In (2), the norm is the ℓ^2 norm, ρ_i is a weighting assigned to the fit to the ith example response, and $\hat{h}_i(p_1, \ldots, p_N)$ minimizes the fit to h_i over choice of system with N poles fixed at $z = p_1, \ldots, p_N$.

D. Transformation of I-O responses into state space descriptions

This transformation is exactly the core of the study about system identification. Many books and articles have been written about the subject, allowing different ways to solve the problem (for example, [9, 13]). Here we use one popular method applicable to the output error formulation: the Steiglitz and McBride technique [15] for the identification of linear systems described by difference equations. Reference [16] shows that if the additive output noise is white and the number of samples is large, then any estimate to which Steiglitz-McBride converges gives a correct description of the system transfer function. Besides this, if the signal-to-noise ratio is sufficiently large or if the system has only one pole, then it is provable that the estimates will converge to the true parameter vector for any arbitrary but stable initialization.

E. Approximation of impulse responses by state space description

Reference [1] presents an algorithm to approximate an FIR filter by an IIR filter. It circumvents the errors associated with matrix inversions when balanced reduction is applied directly, employing a unique property of the FIR filter, and reducing the system order without calculating explicitly its balanced realization. This algorithm has the following steps:

1. Use the given desired FIR filter (or impulse response) coefficients $h(0), h(1), \ldots, h(k)$ to create the Hankel matrix
$$H = \begin{bmatrix} h(0) & h(1) & \cdots & \cdots & h(k) \\ h(1) & h(2) & \cdots & h(k) & 0 \\ \vdots & \vdots & \ddots & & \vdots \\ h(k) & 0 & \cdots & \cdots & 0 \end{bmatrix}.$$
2. Find the singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_k$ and correspondingly ordered eigenvectors $V = [v_1\ v_2\ \cdots\ v_k]$ of H. Choose M, the order of the approximation, based on the relative sizes of $\{\sigma_i\}$.
3. Calculate the reduced, Mth order, state space description A, b, c, and d using
$$\begin{aligned} A &= V(2:k, 1:M)^T V(1:k-1, 1:M) \\ b &= V(1, 1:M)^T \\ c &= CV(1:k, 1:M) \quad \text{where } C = [h(1)\ h(2)\ \cdots\ h(k)] \\ d &= h(0) \end{aligned}$$
where $V(i:j, k:m)$ denotes an extraction of rows i through j and columns k through m from matrix V.

F. Reduction of the number of impulse responses

Reference [21] describes a method to reduce in an optimal fashion M example ideal impulse responses $\{h_i(n)\}_{i=1}^{M}$ to a smaller number N. The technique fits an optimal N-dimensional vector space of systems to the collection of M impulse responses in the following sense. The N impulse response $\{\hat{h}_i(k)\}_{i=1}^{N}$ which define a basis for this vector space are chosen to minimize the error expression
$$\sum_{i=1}^{M} \min_{\alpha_1,\ldots,\alpha_N} \sum_{n=0}^{\infty} \left(h_i(n) - \sum_{j=1}^{N} \alpha_j \hat{h}_j(n) \right)^2. \tag{3}$$
Minimizing this error expression minimizes the total MSE fit between the example filters and the individual, optimal members of the vector space of systems spanned by $\{\hat{h}_i(\cdot)\}$ when the input signal is white. In [21], formulation of a similar optimization problem for colored input signals is also addressed.

4 Examples

We have chosen two different examples to provide understanding about the steps to assemble a FPAF tuned for one application. In each example, we compare several methods of selecting the pole locations, and the performance of the FPAF with adaptive filters based on FIR and Kautz models. These examples use the adaptive filter configuration depicted in Fig. 1, where the performance criterion is the relative error
$$-10 \log \left(\frac{\sum_{i=0}^{M-1} \sum_{n=0}^{L-1} [d_i(n) - y_i(n)]^2}{\sum_{i=0}^{M-1} \sum_{n=0}^{L-1} [d_i(n)]^2} \right). \tag{4}$$
assuming a white input sequence $\{u(n)\}$. M is the number of ideal responses tested, L is the number of measured data points, and $d_i(n)$ is the output of the ith ideal response with $y_i(n)$ the response of the FPAF which best fits d_i in a mean square sense (after the pole locations have been chosen and fixed).

	Type of System Response		
	Impulse		White Noise
No. of Poles	No noise	With Noise	With Noise
6	17.0	17.0	17.1
10	29.7	29.7	29.7
14	43.1	38.4	43.0
18	55.1	48.1	55.1

Table 1: Performance using Method (C), Measured in dB

	Type of System Response		
	Impulse		White Noise
No. of Poles	No noise	With Noise	With Noise
6	18.0	17.9	18.0
10	30.5	30.8	30.5
14	42.7	42.3	42.7
18	54.6	46.4	54.6

Table 2: Performance using Method (B), Measured in dB

4.1 Example 1: Bandpass Butterworth Filters

We randomly select 16 example systems and 50 true systems from a set of bandpass Butterworth transfer functions, whose defining parameters are uniformly distributed between the following limits:

number of poles $\in [6, 12]$
normalized lowpass frequency $\in [0.3\pi, 0.45\pi]$
normalized highpass frequency $\in [0.5\pi, 0.6\pi]$.

Next, we apply several procedures to find the pole locations, with the *a priori* information degraded by noise. To the impulse responses, we use method (C) to find N pole locations. To the white noise responses, we first estimate state space descriptions with 22 states, using Steiglitz-McBride, and we find the impulse responses for each description. Now, we branch again from these M estimate impulse responses to N pole locations, using again (C). To compare the performance of these two ways of pole selection, we generate one more set of poles, but now using pure impulse responses of the example system, without noise added to the output. So, we have three sets of poles, and we adapt the three FPAFs to each of the 50 true systems, and we measure the average squared output error. Table 1 depicts the result of these simulations. One can observe that the method that used the white noise responses and pure impulse responses have the same performance, even for FPAFs with more than 10 poles.

Now, we repeat similar steps as described above, but using method (B) to find N locations. Table 2 shows the results of this experiment. One can observe that the performance is similar to that using method (C) from Table 1.

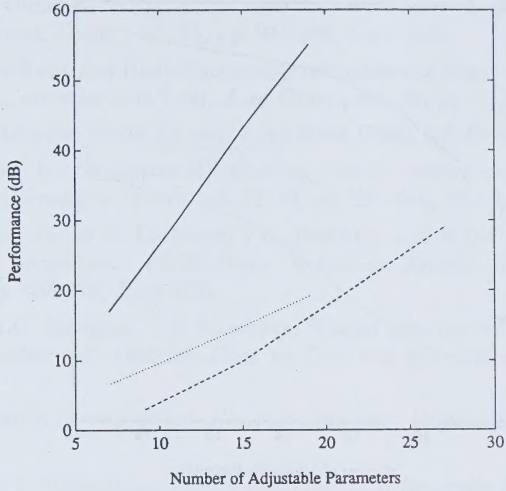


Figure 3: Performance for Butterworth Example. FPAF (solid line), FIR (dashed line), and Kautz (dotted line).

The overall conclusion of these two experiments is that the methods for fixed pole selection perform more poorly in the presence of noise if the *a priori* information is impulse responses rather than white noise responses.

Our next goal is to compare the FPAF performance with that of an FIR and Kautz adaptive filter, when pure impulse responses are used to define the pole locations for the FPAF. For the Kautz filter, we apply method (C) to fit two complex pole locations to the collection of example ideal filters. As a basis for comparison, we use the number of adjusted parameters in each filter. Figure 3 shows the performance of each adaptive filter as a function of this number. One can see that the performance of the FPAF is superior when compared with FIR and Kautz filters, though a better procedure to select the complex conjugate poles for the Kautz filter should be able to improve its performance somewhat.

4.2 Example 2: Echo Cancellation in Long Distance Telecommunications

We have available five impulse responses of typical echo paths in long distance telecommunications connections. Each response has a length of 256 samples, with a typical shape consisting of an initial delay followed by a rapid time variation, and then a slow decay toward zero. Reference [2] suggests a two-stage approach for this type of response: an FIR first stage to capture the delay, and a second stage using Laguerre or Legendre filters. We follow the same approach, modeling the first 50 samples with an FIR filter, but using FPAF, Kautz, or FIR filters to model the remaining samples in the second stage.

We use method (B) from Section 3 to fit a given number of pole locations to the tails of each of the five sample responses, and then fit the resulting FPAF to each response individually. Likewise, we use method (B) to fit two complex pole locations to form a

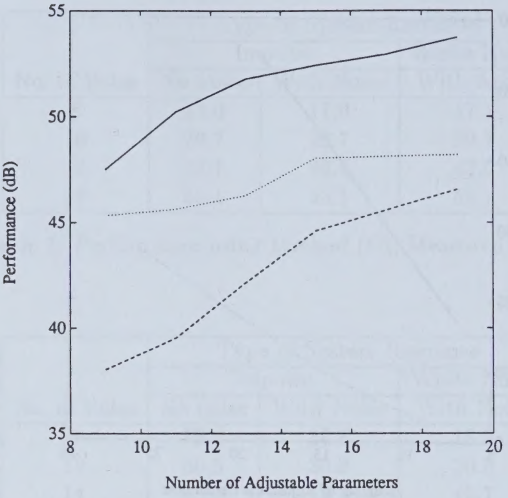


Figure 4: Performance for Telephone Example. FPAF (solid line), FIR (dashed line), and Kautz (dotted line).

Kautz filter description, and then adjust the adaptive Kautz filter to best match each individual response. The same is done for an adaptive FIR filter. The average performances over the five examples, as measured via (4), are plotted in Fig. 4. We can see that the FPAF approach enables a much better performance for a given number of adjustable parameters.

5 Conclusions

This paper has presented an adaptive filtering structure, the FPAF, which may be seen as a compromise between finite impulse response and pole/zero adaptive filters, and as a generalization of the identifier structures using Laguerre, Kautz, and related models. We have discussed a number of techniques for selecting the poles in the FPAF to tailor its specification to the class of unknown ideal filters in any given application. The experiments we reported show improved MSE performance for FPAFs in comparison to FIR and Kautz filters.

References

[1] B. Beliczynski, I. Kale, G.D. Cain, "Approximation of FIR by IIR digital filters: an algorithm based on balanced model reduction," *IEEE Trans. Signal Processing*, vol. 40, pp. 532-541, March 1992.

[2] G.W. Davidson and D.D. Falconer, "Reduced complexity echo cancellation using orthonormal functions," *IEEE Trans. Circuits Syst.*, vol. 38, pp. 20-28, January 1991.

[3] Y. Fu, G.A. Dumont, "An optimum time scale for discrete Laguerre network," *IEEE Trans. Automat. Contr.*, vol. 38, pp. 934-938, June 1993.

[4] K. Glover, "All optimal Hankel norm approximations of linear multivariable systems and their L_∞ error bounds," *Int. J. of Contr.*, vol. 39, pp. 1115-1193, 1984.

[5] S. Haykin, *Adaptive Filter Theory.* Englewood Cliffs, NJ: Prentice-Hall, 1991.

[6] C.R. Johnson, Jr., "Adaptive IIR filtering: current results and open issues," *IEEE Trans. on Information Theory*, vol. IT-30, pp. 237-250, March 1984.

[7] C.R. Johnson, Jr., M.G. Larimore, J.R. Treichler, and B.D.O. Anderson, "SHARF convergence properties," *IEEE Trans. Acoustics, Speech, Signal Processing*, vol. ASSP-29, pp. 659-670, June 1981.

[8] A. Kaelin, A.G. Lindgren, G.S. Moschytz, "Linear echo cancellation using optimized recursive prefiltering," *1993 Int. Conf. on Circ. and Sys.*, Chicago IL, May 1993, pp. 463-466.

[9] L. Ljung, *System Identification: Theory for the User.* Englewood Cliffs, NJ: Prentice-Hall, 1987.

[10] L. Ljung and T. Söderström, *Theory and Practice of Recursive Identification.* Boston: MIT Press, 1983.

[11] H. Perez and S. Tsujii, "A system identification algorithm using orthogonal functions," *IEEE Trans. Sig. Proc.*, vol. SP-39, pp. 752-755, Mar. 1991.

[12] J.C. Principe, B. de Vries, and P.G. de Oliveira, "The gamma filter – a new class of adaptive IIR filters with restricted feedback," *IEEE Trans. Sig. Proc.*, vol. 41, pp. 649-656, Feb. 1993.

[13] P. Stoica, T. Söderström, *System Identification.* Englewood Cliffs, NJ: Prentice-Hall, 1987.

[14] S. Stearns, "Error surfaces of recursive adaptive filters," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-29, pp. 763-766, June 1981.

[15] K. Steiglitz and L.E. McBride, "A technique for identification of linear systems," *IEEE Trans. Automat. Contr.*, vol. AC-10, pp. 461-464, 1965.

[16] P. Stoica, T. Söderström, "The Steiglitz-McBride identification algorithm revisited – convergence analysis and accuracy aspects," *IEEE Trans. Auto. Contr.*, vol. AC-26, June 1981.

[17] B. Wahlberg, "System identification using Laguerre models," *IEEE Trans. Automat. Contr.*, vol. AC-36, pp. 551-562, May 1991.

[18] B. Wahlberg, "Identification of resonant systems using Kautz filters," *Proc. 30th Conf. on Dec. and Contr.*, Brighton, England, Dec. 1991, pp. 2005-2010.

[19] B. Widrow and S. Stearns, *Adaptive Signal Processing*, Prentice-Hall, 1985.

[20] G.A. Williamson, "Adaptive filtering using vector spaces of systems," *1992 IEEE Digital Signal Processing Workshop*, Starved Rock State Park, IL, September 1992.

[21] G.A. Williamson, "Linear-in-the-parameters identification for classes of systems," *1993 Conf. on Dec. and Contr.*, San Antonio TX, Dec. 1993.

[22] S. Zimmermann and G.A. Williamson, "Performance properties of fixed pole adaptive filters," *1993 Int. Conf. on Circ. and Sys.*, Chicago IL, May 1993, pp. 56-59.

A High-Performance Cost-Effective Pole-Zero MMSE-DFE*

N. Al-Dhahir, A. Sayed and J. Cioffi
Information Systems Laboratory
Stanford University, Stanford CA 94305

Abstract

In this paper, we derive a computationally-efficient algorithm for accurately approximating a long FIR filter by a reduced-parameter pole-zero filter. Our derivation successfully extends the "embedding" approach of [1, 2] to the case of an unequal number of poles and zeros.

Our main emphasis is on applying the algorithm to reduce the implementation complexity of the long FIR feedforward and feedback filters of the MMSE-DFE encountered in digital subscriber loops. We also describe several other applications where the algorithm leads to a significant reduction in implementation complexity.

1 Introduction

In a variety of applications, the engineer is faced with the task of implementing a long finite impulse response (FIR) filter on a digital signal processor. For example, this long FIR filter could be the feedforward or the feedback filter of a decision feedback equalizer (DFE), an echo canceller (EC), a simulator of a channel impulse response (IR), etc. Moreover, in some situations the underlying true system response is of infinite length (IIR). Assuming it to be FIR (e.g., by truncating up to the most significant N samples) has some advantages : An ease of computation (e.g., through the use of efficient time-domain algorithms such as the Levinson algorithm or frequency-domain algorithms such as the FFT algorithm) and a lower sensitivity to finite-precision effects. However, to achieve satisfactory performance (a high decision-point SNR for the DFE, large echo suppression for the EC, and a low model approximation error for the channel IR), a large number of FIR filter taps is usually needed. This can lead to a prohibitive implementation cost in terms of the increased memory needed to store the filter taps and the large record of previous input samples, in addition to the high processing power required to compute the filter output samples through sum-of-products calculations. This cost even multiplies for high-speed applications and for time-varying environments where the filter taps are frequently updated.

In this paper, we study the generic problem of approximating a long FIR filter by a pole-zero filter with a much smaller total (numerator and denominator) number of coefficients. This problem has been investigated by many researchers in the contexts of IIR digital filter design (see, e.g., [3, 4, 5]), echo cancellation (see, e.g., [6, 7]), and more recently zero-forcing decision feedback equalization in [8]. Our main focus will be on the DFE application. However, our approach differs significantly from that of [8] in several aspects. First, the tail of the long FIR filter was assumed in [8] to be accurately modeled by two poles only. This assumption is specific to the High-Bit-Rate Digital Subscriber Loop (HDSL) environment considered in that paper. Although we shall also use an HDSL channel in our simulations, the algorithm that we shall present is quite general and does not make any such assumptions. Second, pre-cursor ISI was assumed to be negligible in [8], and hence the feedforward filter was assumed to be a short FIR filter. We do not make this assumption either since for higher data rates and less benign channel characteristics, as in the Asymmetric Digital Subscriber Loop (ADSL) environment, the feedforward filter must be very long to achieve satisfactory performance. Therefore, we shall attempt to approximate both the feedforward and feedback filters by pole-zero models. Finally, the DFE coefficients were computed in [8] using adaptive IIR algorithms. Again, the environment-specific assumption of a 2-pole model made stability monitoring a simple task, which would not be the case in situations where a 2-pole model is not adequate (e.g. in echo cancellation). Instead, we shall compute the DFE coefficients directly from the available channel and noise estimates using the efficient algorithms of [9]. In case of environment changes, straightforward adaptation is performed on the long FIR filter, which is then converted to a pole-zero filter for a reduced-complexity implementation.

---

*This work was supported by NSF contract number NCR-9203131, JSEP contract number DAAL03-91-C-0010, NASA contract number NAG2-842.

2 Pole-Zero Modeling of a Long FIR Filter

2.1 Motivation

The FIR MMSE–DFE is a widely–used receiver structure that mitigates the effects of severe ISI and noise to restore communication integrity. As shown in Figure 1, it consists of two FIR filters. A feedforward filter $w(D)$, with N_f taps $\{w_{-\Delta}, \cdots, w_{N_f-1-\Delta}\}$ (where Δ is the decision delay) that combats pre–cursor ISI and noise, and a strictly–causal feedback filter $b(D)$ with N_b taps $\{-b_1, \cdots, -b_{N_b}\}$ that suppresses post–cursor ISI.

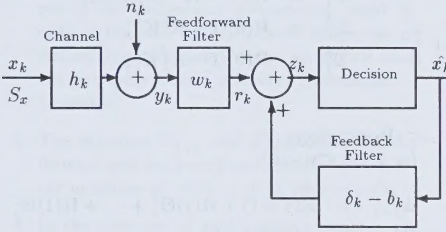


Figure 1: Block diagram of the FIR MMSE–DFE

The channel pulse response $h(D)$ is assumed to be a linear time–invariant FIR filter with a memory of ν, i.e., $h(D) = h_0 + h_1 D + \cdots + h_\nu D^\nu$.

Closed–form expressions for the optimal feedforward and feedback filter settings that minimize the mean square error $MSE \overset{def}{=} E|e_k|^2 = E|x_{k+N_f-1-\Delta} - z_k|^2$ were derived in [10] for arbitrary choices of N_f and N_b. However, it was shown in [9] that *significant* computational reductions can be achieved when we assume that the number of feedback taps is equal to the channel memory, i.e., $N_b = \nu$. Under this assumption, the feedforward filter becomes strictly anti–causal and $\Delta = N_f - 1$. More specifically, assuming white input with average energy of S_x per complex dimension and correlated–noise with a (non–singular) correlation matrix $\mathbf{R}_{nn}$, then the optimal FIR MMSE–DFE can be computed efficiently from the Cholesky factorization :

$$\frac{1}{S_x}\mathbf{I}_{N_f+\nu} + \mathbf{H}^*\mathbf{R}_{nn}^{-1}\mathbf{H} \overset{def}{=} \mathbf{LDL}^* \qquad (1)$$

as follows :

$$\begin{aligned} \mathbf{b}_{opt.} &= \mathbf{Le}_{N_f} & (2)\\ \mathbf{w}^*_{opt.} &= d^{-1}_{N_f-1}(\mathbf{e}^*_{N_f}\mathbf{L}^{-1})\mathbf{H}^*, & (3) \end{aligned}$$

where $\mathbf{I}_{N_f+\nu}$ is the identity matrix of size $N_f+\nu$, $(.)^*$ denotes the conjugate transpose, $\mathbf{e}_{N_f}$ is the N_f^{th} unit vector, and $\mathbf{H}$ is the $N_f \times (N_f+\nu)$ fully–windowed Toeplitz matrix given by

$$\mathbf{H} = \begin{bmatrix} h_0 & h_1 & \cdots & h_\nu & 0 & \cdots & 0 \\ 0 & h_0 & h_1 & \cdots & h_\nu & 0 & \cdots \\ \vdots & & & & & & \vdots \\ 0 & \cdots & 0 & h_0 & h_1 & \cdots & h_\nu \end{bmatrix}. \qquad (4)$$

In many applications (such as high–speed data transmission on twisted copper lines that will be discussed in more detail in Section 3), ν is very large, which entails the use of very long feedforward and feedback filters (a total of more than a hundred taps) to achieve satisfactory performance. Although these long filters can be computed very efficiently using (1)-(3), implementing them in real–time is very costly. This consideration has motivated us to develop an efficient algorithm to convert long FIR filters to pole–zero filters with much fewer parameters, without loosing stability, and while still maintaining satisfactory performance.

Another requirement on this algorithm is the ability to handle mixed–phase FIR filters. This is due to the fact shown in [10] that the optimal feedforward (feedback) filters of the FIR MMSE–DFE are not necessarily maximum phase (minimum phase), *unless N_f is infinite.*

2.2 The Generalized ARMA-Levinson Algorithm

In this section, we shall derive a *new* algorithm for approximating a long FIR filter by a pole–zero *stable* filter with much fewer taps. The algorithm is a generalization of the ARMA–Levinson algorithm derived in [2] using the "embedding" technique of [1]. The novelty in our algorithm is *its ability to relax the restriction of an equal number of poles and zeros* that was assumed in [2, 1]. This flexibility will prove to be useful in obtaining better fits, as it will become clear from the simulation results of Section 3.

An ARMA model with q zeros and p poles, denoted in this paper by ARMA(p,q), is described by the following difference equation : [1]

$$y_k = -a_1 y_{k-1} - \cdots - a_p y_{k-p} + n_0 x_k + \cdots + n_q x_{k-q}\,. \qquad (5)$$

Assume first a *proper* transfer function so that $p \geq q$. The case of $q > p$ requires some modifications and will be described later.

Our objective is to estimate the $(p+q+1)$ ARMA parameters $\{a_1, \cdots, a_p, n_0, \cdots, n_q\}$ based on knowledge of the second–order statistics of the input and output sequences. If we denote these estimates at the j^{th} recursion ($1 \leq j \leq p$) by $\{a_1^j, \cdots, a_j^j, n_0^j, \cdots, n_{j-\delta}^j\}$ where $\delta \overset{def}{=} p - q \geq 0$, then the output of this ARMA$(j, j-\delta)$ filter can be calculated as follows :

---

[1] In the sequel, the numerator polynomial coefficients $\{n_0, \cdots, n_q\}$ are not to be confused with the noise samples n_k. Distinction should be clear from the context.

$$\begin{aligned} y_k &= -a_1^j y_{k-1} - \cdots - a_j^j y_{k-j} + n_0^j x_k + \cdots \\ &+ n_{j-\delta}^j x_{k-j+\delta} + e_k^j \end{aligned} \tag{6}$$

where e_k^j is the j^{th} order residual error sequence.

If we define the augmented vector $\mathbf{z}_k \overset{def}{=} \begin{bmatrix} y_k \\ x_{k+\delta} \end{bmatrix}$, then the ARMA model of (6) can be converted to the 2–channel AR model :

$$\begin{aligned} &\begin{bmatrix} y_k \\ x_{k+\delta} \end{bmatrix} = \begin{bmatrix} -a_1^j & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} y_{k-1} \\ x_{k+\delta-1} \end{bmatrix} + \cdots \\ &+ \begin{bmatrix} -a_\delta^j & n_0^j \\ 0 & 0 \end{bmatrix} \begin{bmatrix} y_{k-\delta} \\ x_k \end{bmatrix} + \begin{bmatrix} -a_{\delta+1}^j & n_1^j \\ 0 & 0 \end{bmatrix} \begin{bmatrix} y_{k-\delta-1} \\ x_{k-1} \end{bmatrix} \\ &+ \cdots + \begin{bmatrix} -a_j^j & n_{j-\delta}^j \\ 0 & 0 \end{bmatrix} \begin{bmatrix} y_{k-j} \\ x_{k-j+\delta} \end{bmatrix} + \begin{bmatrix} e_k^j \\ x_{k+\delta} \end{bmatrix}, \end{aligned} \tag{7}$$

or more compactly,

$$\mathbf{z}_k = -\boldsymbol{\Theta}_1^j \mathbf{z}_{k-1} - \cdots - \boldsymbol{\Theta}_j^j \mathbf{z}_{k-j} + \begin{bmatrix} e_k^j \\ x_{k+\delta} \end{bmatrix}, \tag{8}$$

where we have defined

$$-\boldsymbol{\Theta}_i^j \overset{def}{=} \begin{cases} \begin{bmatrix} -a_i^j & 0 \\ 0 & 0 \end{bmatrix} & : 1 \leq i \leq \delta - 1 \\ \begin{bmatrix} -a_i^j & n_{i-\delta}^j \\ 0 & 0 \end{bmatrix} & : \delta \leq i \leq j \end{cases} \tag{9}$$

If we multiply both sides of (8) by $\mathbf{z}_{k-i}^*$ ($0 \leq i \leq j$) and take expectations, we arrive at

$$\mathbf{R}(0) + \boldsymbol{\Theta}_1^j \mathbf{R}(-1) + \cdots + \boldsymbol{\Theta}_j^j \mathbf{R}(-j) = \boldsymbol{\Sigma}_j^f \ : \ i = 0 \tag{10}$$

$$\mathbf{R}(i) + \boldsymbol{\Theta}_1^j \mathbf{R}(i-1) + \cdots + \boldsymbol{\Theta}_j^j \mathbf{R}(i-j) = 0 \ : \ 1 \leq i \leq j \tag{11}$$

where

$$\begin{aligned} \mathbf{R}(i) &\overset{def}{=} E[\mathbf{z}_k \mathbf{z}_{k-i}^*] = \begin{bmatrix} R_{yy}(i) & R_{yx}(i-\delta) \\ R_{xy}(i+\delta) & R_{xx}(i) \end{bmatrix} \\ &= \mathbf{R}^*(-i) . \end{aligned} \tag{12}$$

Alternatively, (10) and (11) can be written in matrix form as follows :

$$\boldsymbol{\Theta}^j \mathbf{R}^j = \begin{bmatrix} \mathbf{0} & \cdots & \mathbf{0} & \boldsymbol{\Sigma}_j^f \end{bmatrix}, \tag{13}$$

where

$$\begin{aligned} \boldsymbol{\Theta}^j &= \begin{bmatrix} \boldsymbol{\Theta}_j^j & \cdots & \boldsymbol{\Theta}_1^j & \mathbf{I} \end{bmatrix} \\ \mathbf{R}^j &= \begin{bmatrix} \mathbf{R}(0) & \cdots & \mathbf{R}(-(j-1)) & \mathbf{R}(-j) \\ \vdots & \ddots & \vdots & \vdots \\ \mathbf{R}(j-1) & \cdots & \mathbf{R}(0) & \mathbf{R}(-1) \\ \mathbf{R}(j) & \cdots & \mathbf{R}(1) & \mathbf{R}(0) \end{bmatrix} \end{aligned}$$

Equation (13) describes a j^{th}–order AR model whose vector parameters $\boldsymbol{\Theta}_i^j$ can be calculated by solving a block–Toeplitz Hermitian system of linear equations. This can be done efficiently using the following multichannel form of the scalar Levinson Algorithm, sometimes known as the Levinson–Wiggins–Robinson (LWR) Algorithm[11, 12].

Algorithm 1 (Generalized ARMA-Levinson)
Given $\{\mathbf{R}(0), \cdots, \mathbf{R}(p)\}$

Initial Conditions :

$$\begin{aligned} \boldsymbol{\Theta}_1^1 &= \mathbf{K}_1^f = -\mathbf{R}^{-1}(0)\mathbf{R}(1) . \\ \boldsymbol{\Phi}_1^1 &= \mathbf{K}_1^b = -\mathbf{R}^{-1}(0)\mathbf{R}(-1) . \\ \boldsymbol{\Sigma}_1^f &= \mathbf{R}(0)(\mathbf{I} - \mathbf{K}_1^b \mathbf{K}_1^f) . \\ \boldsymbol{\Sigma}_1^b &= \mathbf{R}(0)(\mathbf{I} - \mathbf{K}_1^f \mathbf{K}_1^b) . \end{aligned}$$

Recursions :
for $1 \leq j \leq p-1$

$$\begin{aligned} \boldsymbol{\Delta}_{j+1}^f &= \mathbf{R}(j+1) + \mathbf{R}(j)\boldsymbol{\Theta}_1^j + \cdots + \mathbf{R}(1)\boldsymbol{\Theta}_j^j . \\ \mathbf{K}_{j+1}^f &= -(\boldsymbol{\Sigma}_j^b)^{-1} \boldsymbol{\Delta}_{j+1}^f . \\ \mathbf{K}_{j+1}^b &= -(\boldsymbol{\Sigma}_j^f)^{-1} \boldsymbol{\Delta}_{j+1}^{*f} . \\ \boldsymbol{\Sigma}_{j+1}^f &= \boldsymbol{\Sigma}_j^f (\mathbf{I} - \mathbf{K}_{j+1}^b \mathbf{K}_{j+1}^f) . \\ \boldsymbol{\Sigma}_{j+1}^b &= \boldsymbol{\Sigma}_j^b (\mathbf{I} - \mathbf{K}_{j+1}^f \mathbf{K}_{j+1}^b) . \\ \boldsymbol{\Theta}_i^{j+1} &= \boldsymbol{\Theta}_i^j + \boldsymbol{\Phi}_{j-i+1}^j \mathbf{K}_{j+1}^f \ : \ 1 \leq i \leq j . \\ \boldsymbol{\Theta}_{j+1}^{j+1} &= \mathbf{K}_{j+1}^f . \\ \boldsymbol{\Phi}_i^{j+1} &= \boldsymbol{\Phi}_i^j + \boldsymbol{\Theta}_{j-i+1}^j \mathbf{K}_{j+1}^b \ : \ 1 \leq i \leq j . \\ \boldsymbol{\Phi}_{j+1}^{j+1} &= \mathbf{K}_{j+1}^b , \end{aligned} \tag{14}$$

where the backward prediction vector $\boldsymbol{\Phi}^j \overset{def}{=} \begin{bmatrix} \mathbf{I} & \boldsymbol{\Phi}_1^j & \cdots & \boldsymbol{\Phi}_j^j \end{bmatrix}$ satisfies the following auxiliary block–Toeplitz system of equations :

$$\boldsymbol{\Phi}^j \mathbf{R}^j = \begin{bmatrix} \boldsymbol{\Sigma}_j^b & \mathbf{0} & \cdots & \mathbf{0} \end{bmatrix} .$$

Assuming the input sequence to be *white* (i.e., $R_{xx}(l) = S_x \delta_l$), then the output autocorrelation and input–output cross–correlation sequences needed to form (12) can be computed using knowledge of the FIR filter taps as follows

$$R_{yy}(l) = S_x \sum_{m=0}^{\nu} h_m h_{m-l}^* . \tag{15}$$

$$R_{yx}(l) = S_x h_l = R_{xy}^*(-l) . \tag{16}$$

The parameters $\{a_1^p, \cdots, a_p^p, n_0^p, n_1^p, \cdots, n_q^p\}$ are read off directly from $\boldsymbol{\Theta}_i^p$ ($1 \leq i \leq p$).

Remarks :

1. For the special case of an equal number of poles and zeros, we have $\delta = 0$. Hence, the first δ terms of (7) disappear, i.e.,
$$\begin{bmatrix} y_k \\ x_k \end{bmatrix} = \begin{bmatrix} -a_1^j & n_1^j \\ 0 & 0 \end{bmatrix} \begin{bmatrix} y_{k-1} \\ x_{k-1} \end{bmatrix} + \cdots$$
$$+ \begin{bmatrix} -a_j^j & n_j^j \\ 0 & 0 \end{bmatrix} \begin{bmatrix} y_{k-j} \\ x_{k-j} \end{bmatrix} + \begin{bmatrix} 1 & n_0^j \\ 0 & 1 \end{bmatrix} \begin{bmatrix} e_k^{'j} \\ x_k \end{bmatrix} . \tag{17}$$
Algorithm 1 can still be applied (with $\delta = 0$) to compute $\{a_1^p, \cdots a_p^p, n_1^p, \cdots, n_p^p\}$ and $n_0^p = \mathbf{\Sigma}_p^f(1,2)$.

2. Although our algorithm uses only second–order statistics, it can generate non–minimum phase approximations (i.e., the zeros of the numerator polynomial could lie outside the unit circle) because of the use of embedding. However, it can be shown that the pole polynomial is always guaranteed to be stable.

3. The matrices $\mathbf{\Sigma}_{j+1}^f$ and $\mathbf{\Sigma}_{j+1}^b$ are called the forward and backward prediction residual error matrices of order $(j+1)$, respectively.

4. In the presence of additive noise that is *independent of the input* with a correlation sequence $R_{nn}(l)$, Equation (15) is modified to : $R_{yy}(l) = S_x \sum_{m=0}^{\nu} h_m h_{m-l}^* + R_{nn}(l)$.

5. The case $q > p$ can be handled by interchanging the roles of the input and output sequences in (6). More specifically, consider the following ARMA(q,p) model of the *inverse filter*:
$$\begin{aligned} x_k &= -n_1^j x_{k-1} - \cdots - n_j^j x_{k-j} + a_0^j y_k + \cdots \\ &\quad + a_{j-\delta}^j y_{k-j+\delta} + e_k^{''j} , \end{aligned} \tag{18}$$
where $\delta \overset{def}{=} q - p$ and $1 \leq j \leq q$. Then, the embedding relation of (7) is modified to
$$\begin{bmatrix} x_k \\ y_{k+\delta} \end{bmatrix} = \begin{bmatrix} -n_1^j & 0 \\ \times & \times \end{bmatrix} \begin{bmatrix} x_{k-1} \\ y_{k+\delta-1} \end{bmatrix} + \cdots$$
$$+ \begin{bmatrix} -n_\delta^j & a_0^j \\ \times & \times \end{bmatrix} \begin{bmatrix} x_{k-\delta} \\ y_k \end{bmatrix}$$
$$+ \begin{bmatrix} -n_{\delta+1}^j & a_1^j \\ \times & \times \end{bmatrix} \begin{bmatrix} x_{k-\delta-1} \\ y_{k-1} \end{bmatrix}$$
$$+ \cdots + \begin{bmatrix} -n_j^j & a_{j-\delta}^j \\ \times & \times \end{bmatrix} \begin{bmatrix} x_{k-j} \\ y_{k-j+\delta} \end{bmatrix} + \begin{bmatrix} e_k^{''j} \\ y_{k+\delta} \end{bmatrix} . \tag{19}$$
Define $\mathbf{R}(i) \overset{def}{=} \begin{bmatrix} R_{xx}(i) & R_{xy}(i-\delta) \\ R_{yx}(i+\delta) & R_{yy}(i) \end{bmatrix}$, then Algorithm 1 can be applied to compute the parameters of the ARMA model $h(D) \approx \frac{1+n_1^q D + \cdots + n_q^q D^q}{a_0^q + a_1^q D + \cdots + a_{q-\delta}^q D^{q-\delta}}$.

6. A useful way to interpret Algorithm 1 is as follows : we apply the $\mathbf{\Theta}_i^{j+1}$ recursion in (14) only for $\delta \leq i \leq j$. The parameters $\{\mathbf{\Theta}_1^{\delta-1}, \cdots, \mathbf{\Theta}_{\delta-1}^{\delta-1}\}$, or equivalently the all–pole prediction coefficients $\{a_1^{\delta-1}, \cdots, a_{\delta-1}^{\delta-1}\}$ can be computed by applying the classical (scalar) Levinson Algorithm. That is, we first find the best δ^{th} order AR model and then use it for the initialization of Algorithm 1 that will successively generate the best ARMA$(i+\delta, i)$ models $(1 \leq i \leq q)$, until it finally arrives at the desired ARMA$(q+\delta, q)$ model.

7. The formulation of (7) accommodates the more general case where the input sequence is also modeled as an ARMA process. Our assumption of a white input sequence is a special case of that general formulation.

8. It was brought to our attention recently [2] that a multichannel least squares algorithm with a different number of parameters to be estimated in each channel was previously derived in [13]. However, the algorithm of [13] is distinct from ours in that it is of lattice type, time–recursive, and was not derived using the "embedding" approach that we follows here. In addition, the intended application of this algorithm being investigated in this paper, namely pole–zero modeling of long FIR filters, is new, to the best of our knowledge.

2.3 Model Order Selection

A critical component in generating an accurate ARMA(p,q) approximation is the choice of p and q. Several tests have been proposed in the literature for determining p and q from the output data record or from the output auto–correlation sequence [11] (which can be computed from (15) using knowledge of the FIR filter taps and assuming a white input sequence). These tests include an information–theoretic criterion (AIC) by Akaike [14] and singularity tests of an output auto–correlation matrix [15]. More recently, Pillai *et al.* proposed an order–determining method based on a degree–reducing procedure from passive network theory [16].

For the purposes of this paper, we shall choose p and q according to the following two considerations. The first consideration is cost–driven in that we assume a maximum allowable implementation complexity, which in turn sets an upper bound on the values of p and q. Therefore, we seek to find the best, in terms of low normalized norm tap error NNTE $\overset{def}{=} \frac{\sum_{i=0}^{N-1} |h_i - \hat{h}_i|^2}{\sum_{i=0}^{N-1} |h_i|^2}$ (where N is the number of FIR filter taps), pole–zero approximation, subject to this complexity constraint. It is worth emphasizing that increasing p and/or q could result in a worse approximation depending on the FIR filter response, as it will be shown in Section 3. The second consideration is performance–driven and requires an exhaustive search over different choices of p and q, within the allowable range determined by the first consideration above, until we arrive at an acceptable fit. Depending on the application, complexity might be partially traded for performance (as in echo cancellation) or vice versa, although in most cases a compromise between the two is more desirable.

---

[2] Thanks to Professor H. Lev-Ari of Northeastern University.

3 Simulation Results

In this section, we shall evaluate the performance of Algorithm 1 by applying it to a representative example from the HDSL environment to implement the FIR MMSE–DFE filters in a pole–zero form that reduces complexity while still retaining satisfactory performance.

Approximating the feedforward filter :

As mentioned previously in Section 2.1, when $N_b = \nu$, the optimal feedforward filter of the FIR MMSE–DFE becomes strictly anti–causal and can be realized with a delay of $(N_f - 1)$ symbol periods, i.e.,

$$w^*(D^{-1}) = w^*_{-(N_f-1)} D^{-(N_f-1)} + \cdots + w^*_{-1} D^{-1} + w^*_0 \,. \tag{20}$$

We have found through extensive computer simulations that a direct reduced–parameter pole–zero approximation of $D^{N_f} w^*(D^{-1})$ is very difficult to obtain. This is due to the fact that the initial part of the impulse response (IR) prior to the peak could be very long. Pole–zero models can more easily model the decaying tail of the IR following the peak. With this observation in mind, we propose the following procedure for deriving pole–zero models of (20). Denote the peak sample of the feedforward filter IR by $w^*_{-\tau}$ $(0 \leq \tau \leq N_f - 1)$. Shift the feedforward IR by τ samples to center the peak sample at the origin. The resulting IR, namely $D^\tau w^*(D^{-1})$, will have a causal and a strictly anti–causal components that we shall denote by $w_1(D)$ and $w_2^*(D^{-1})$, respectively.

$$D^\tau w^*(D^{-1}) = w_1(D) + w_2^*(D^{-1}) \,. \tag{21}$$

Now, each of $w_1(D)$ and $w_2(D)$ has the shape of a decaying IR tail that can be easily modeled as a pole–zero filter with few parameters. Therefore, (21) becomes

$$D^\tau w^*(D^{-1}) \approx \frac{n^{w_1}(D)}{a^{w_1}(D)} + \frac{n^{*w_2}(D^{-1})}{a^{*w_2}(D^{-1})} \,,$$

and the causal feedforward filter IR is approximated by

$$D^{N_f-1} w^*(D^{-1}) \approx D^{N_f-1-\tau}\left(\frac{n^{w_1}(D)}{a^{w_1}(D)} + \frac{n^{*w_2}(D^{-1})}{a^{*w_2}(D^{-1})}\right) \tag{22}$$

$$= D^{N_f-1-\tau}\left(\frac{n^{w_1}(D) a^{*w_2}(D^{-1}) + a^{w_1}(D) n^{*w_2}(D^{-1})}{a^{w_1}(D) a^{*w_2}(D^{-1})}\right) \tag{23}$$

Realizing the feedforward as in (22) is preferable over the form of (23) since the former requires a smaller number of taps. This further explains the difficulty encountered when attempting a direct reduced–parameter pole–zero approximation of the overall IR of the feedforward filter.

Example 1 :

For the HDSL environment we shall assume the worst case 9 kft 26 AWG DSL. The input power level is 17 dBm evenly–distributed over the transmission bandwidth and the 2–sided AWGN power spectral density (psd) is taken to be -113 dBm/Hz. The standard $|H_x(f)|^2 = k_{NEXT} f^{\frac{3}{2}}$ near–end crosstalk (NEXT) model is adopted with $k_{NEXT} = 10^{-13}$. The target bit rate is set at 800 kbps and the input signal constellation is 16 QAM which is near–optimum [17]. A fixed probability of error $P_e = 10^{-7}$ is assumed and a 4.2 coding gain is included. The finite–length MMSE–DFE is assumed to have 96 feedforward taps and 64 feedback taps. This choice results in an operational margin of around 4.9 dB [17].

In Tables 1-3 we present some of the best pole–zero approximations of the 64–tap feedback filter and the two components of the 96–tap feedforward filter obtained using Algorithm 1 and their corresponding NNTE's. The impulse responses of the those approximations together with that of the desired response are given in Figures 2,4, and 5.

It is evident that the proposed algorithm generates fairly accurate pole–zero approximations with the added advantages of *significant* reduction in the number of filter coefficients, fast computation, and guaranteed stability.

4 Other Applications

Although our main motivation for deriving Algorithm 1 was to reduce the implementational complexity of the finite–length MMSE–DFE, these

(p,q)	$\hat{b}(D)$	NNTE(dB)
$(3,2)$	$\frac{-1+0.5354D+0.5056D^2}{1-1.3501D+0.2757D^2+.1275D^3}$	-27.2946
$(4,4)$	$\frac{-1+0.6731D+0.7541D^2-.2917D^3-.1145D^4}{1-1.4879D+0.1395D^2+0.5757D^3-.1944D^4}$	-31.1634
$(7,3)$	$\frac{-1+.7325D+.572D^2-.2791D^3}{1-1.5472D+.3699D^2+0.3943D^3-.2299D^4+.0636D^5-.0157D^6+.0048D^7}$	-31.384

Table 1: ARMA approximations for $b(D)$ of Example 1 using Algorithm 1 and their achievable NNTE's

(p,q)	$\hat{w}_1(D)$	NNTE(dB)
$(3,2)$	$\frac{47.1549-6.4777D-18.9055D^2}{1-1.0134D-.127D^2+.207D^3}$	-36.5522
$(5,2)$	$\frac{47.1549-4.3601D-24.7598D^2}{1-.9685D-.2905D^2+.3231D^3-.0263D^4+.0166D^5}$	-41.9531
$(6,2)$	$\frac{47.1549-6.5874D-24.4024D^2}{1-1.0157D-.2416D^2+.3091D^3-.0193D^4+.0131D^5+.0028D^6}$	-42.2905

Table 2: ARMA approximations for $w_1(D)$ of Example 1 using Algorithm 1 and their achievable NNTE's

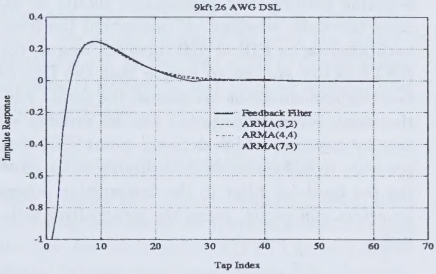


Figure 2: Pole-zero approximations of the feedback filter of Example 1 generated using Algorithm 1.

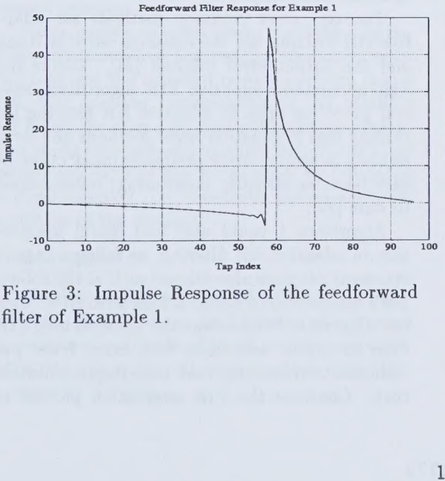


Figure 3: Impulse Response of the feedforward filter of Example 1.

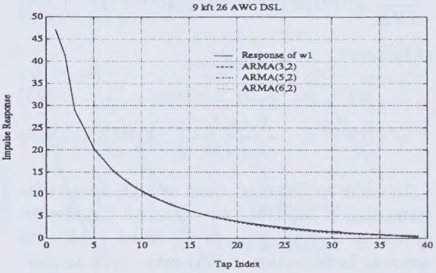


Figure 4: Pole-zero approximations for $w_1(D)$ of Example 1 generated using Algorithm 1.

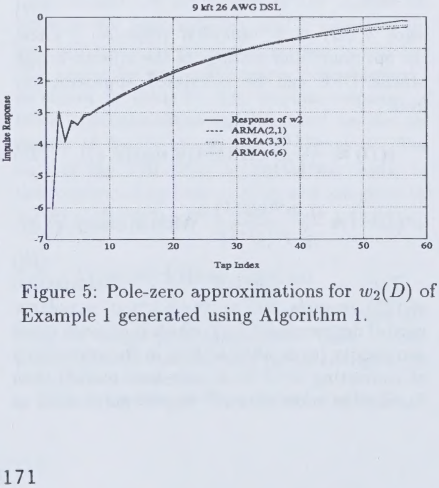


Figure 5: Pole-zero approximations for $w_2(D)$ of Example 1 generated using Algorithm 1.

(p,q)	$\hat{w}_2(D)$	NNTE(dB)
$(2,1)$	$\frac{-6.0337+.4348D}{1-.5621D-.3772D^2}$	-26.0219
$(3,3)$	$\frac{-6.0337+2.7584D+.6662D^2-.3245D^3}{1-.9472D-.2917D^2+.2708D^3}$	-26.4864
$(6,6)$	$\frac{-6.0337+2.5235D+.6947D^2+.7904D^3+.7084D^4+.2405D^5-.1915D^6}{1-.9082D-.3155D^2+.0725D^3-.0224D^4+.0965D^5+.0914D^6}$	-30.4245

Table 3: ARMA approximations for $w_2(D)$ of Example 1 using Algorithm 1 and their achievable NNTE's

algorithms can be used in several other applications. We briefly explore some of these applications next.

4.1 Revisiting the infinite-length MMSE-DFE

It is well known (see e.g. [18, 19]) that the optimum settings of the *infinite-length* MMSE-DFE can be computed from the spectral factorization

$$\frac{1}{SNR} + h(D)h^*(D^{-1}) \overset{def}{=} \gamma_0^2 g(D)g^*(D^{-1}) \tag{24}$$

as follows

$$\begin{aligned} b(D) &= g(D) \ : \nu \text{ taps} && (25) \\ w^*(D^{-1}) &= \frac{1}{\gamma_0^2}\frac{h^*(D^{-1})}{g^*(D^{-1})} : \text{ARMA}(\nu,\nu) && (26) \end{aligned}$$

In some applications, such as high-speed data transmission on DSL, the channel impulse response is very long (large ν), which makes the spectral factorization of (24) very costly to compute. However, if we use Algorithm 1 to convert $h(D)$ to a pole-zero model, $h(D) \approx \frac{n^h(D)}{a^h(D)} \overset{def}{=} \frac{\sum_{i=0}^{q} n_i^h D^i}{1+\sum_{i=1}^{p} a_i^h D^i}$, then (24) becomes

$$\frac{1}{SNR}a^h(D)a^{*h}(D^{-1}) + n^h(D)n^{*h}(D^{-1}) \approx$$

$$\gamma_0^2 g(D)g^*(D^{-1})a^h(D)a^{*h}(D^{-1}) \overset{def}{=} \gamma_0^2 r(D)r^*(D^{-1}) \tag{27}$$

since $a^h(D)$ is a *canonical* response. Then, the optimum filter settings of the infinite-length MMSE-DFE can be calculated approximately from :

$$b(D) \approx \frac{r(D)}{a^h(D)} : \text{ARMA}(p, \max(p,q)) \tag{28}$$

$$w^*(D^{-1}) \approx \frac{1}{\gamma_0^2}\frac{n^{*h}(D^{-1})}{r^*(D^{-1})} : \text{ARMA}(\max(p,q), q)\,. \tag{29}$$

The spectral factorization of (27) requires finding the roots of a polynomial of degree=max(p,q), which is of much lower complexity (even after adding in the complexity of converting $h(D)$ to a pole-zero model) than finding the roots of a ν^{th}-degree polynomial as in (27), since $\nu \gg \max(p,q)$. Nevertheless, we need to take into consideration the degradation of the decision-point SNR when calculating the MMSE-DFE filter settings from (28) and (29) instead of (25) and (26). This issue is addressed in [20].

Another point that deserves attention is the concern that implementing the feedback filter in a pole-zero form might exacerbate the problem of *error propagation*. However, a little reflection shows that when error propagation occurs, the resulting performance degradation should be almost the same whether we implement the feedback filter in an FIR or IIR form, as long as the NNTE is low enough to ensure that the IR's of both implementations are almost the same. Furthermore, error propagation can be avoided either by ensuring a high decision-point SNR that prevents initial errors from occurring or by moving the feedback filter to the transmitter, where no errors can occur, using the **precoding** technique.

4.2 Adaptive IIR filtering

In adaptive filtering applications, the reductions in the implementation cost of pole-zero filters over long FIR filters become more dramatic since a larger memory size and a higher computational power is needed (to store the filter taps and previous input samples and to compute the filter output samples) when the FIR filter coefficients are updated frequently.

The two most common methods for adaptive IIR filtering are the equation-error method and the output-error method [21]. Unlike the well-understood adaptive FIR algorithms, several problems arise in adaptive IIR filtering including bias in equation-error methods and local minima in output-error methods; many other issues (such as stability monitoring) remain open to date [22].

Algorithm 1 could also find useful application in adaptive IIR filtering, as follows. Apply standard adaptive algorithms (such as the robust LMS algorithm) to adapt a long FIR filter, then use Algorithm 1 to convert the updated long FIR filter to a pole-zero filter with much fewer parameters to reduce the real-time implementation cost. Continue the FIR adaptation process in the background and the conversion process in the foreground.

4.3 Echo Cancellation

Echo cancellation for full-duplex transmission on DSL might prove to be one of the most challenging applications for the algorithm of Section 2. The stringent requirement of an NNTE of -60 dB or less in this application makes the pole-zero modeling process more difficult than it is, e.g., in the DFE application. Furthermore, the echo path IR is typically characterized by a rapidly varying initial part followed by a very long tail that could span hundreds of taps, especially as the sampling rate increases.

Additional procedures will be needed to obtain high-quality pole-zero approximations of the echo response. Among those are use of the "splitting" technique (see [7, 8] and Section 4.4) to model the initial part of the echo IR by an FIR filter and only the tail of the IR by a pole-zero filter. We might also need to relax the requirement of $p, q \leq 10$ that was made in the DFE simulations of Section 3, and consider higher-order approximations.

4.4 Channel Identification

Very accurate and efficient channel identification can be carried out in the frequency domain using the Fast Fourier Transform (FFT) algorithm and special periodic training sequences [23]. However, for high spectral resolution, the FFT size N has to be large (typically 512 or 1024) which results in a long N-tap FIR representation of the channel. Algorithm 1 can then be used to convert this long FIR filter to a pole-zero filter.

Direct identification of ARMA(p, q) channel models can also be achieved using Algorithm 1 by computing estimates of the correlation sequences based on known training sequences and measured output samples.

For a variety of other tasks, including transmitter optimization (optimizing the input symbol rate and the transmit power spectrum shape [19, 17]), receiver optimization of the MMSE-DFE (c.f. Section 4.1), and performance evaluation using computer simulations, significant computational savings can be accrued by obtaining a more compact (i.e., has fewer parameters) description of the channel.

Example 2

Consider again the 9 kft 26 AWG DSL of Example 1. In Table 4, we list a number of pole-zero approximations to the channel impulse response, obtained using Algorithm 1, together with their achievable NNTE's.

To illustrate the ability of the algorithm to model non minimum-phase responses such as the channel response of this example (which has 7 non minimum-phase zeros), we have plotted in Figures 6-7 the magnitude and phase responses of the channel and its pole-zero approximations generated using Algorithm 1.

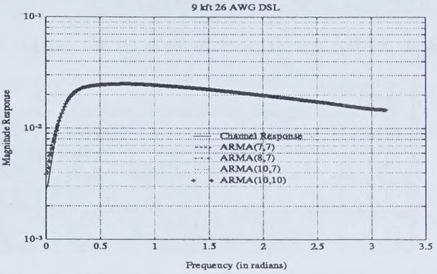


Figure 6: Magnitude response of $h(D)$ of Example 3 and its pole-zero approximations generated using Algorithm 1.

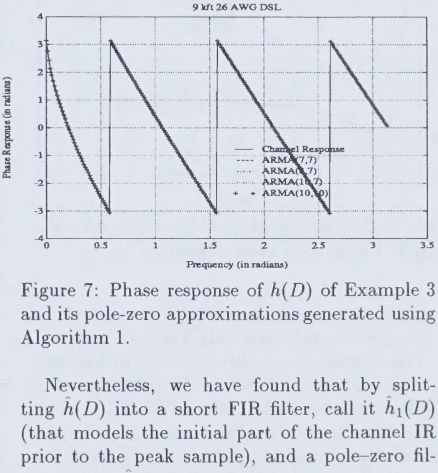


Figure 7: Phase response of $h(D)$ of Example 3 and its pole-zero approximations generated using Algorithm 1.

Nevertheless, we have found that by splitting $\hat{h}(D)$ into a short FIR filter, call it $\hat{h}_1(D)$ (that models the initial part of the channel IR prior to the peak sample), and a pole-zero filter, call it $\hat{h}_2(D)$ (that models the tail of the IR), better fits with lower orders were obtained,[3] as shown in Table 5. The impulse response of those approximations superimposed on the response of $h_2(D)$ are shown in Figure 8. The taps of the FIR filter $\hat{h}_1(D)$ are set equal to the corresponding ones in $h(D)$ and are given by $\hat{h}_1(D) = 10^{-4}(D - D^2 + 2D^3 - 4D^4 + 11D^5)$.

4.5 Digital Filter Design

Recursive digital filters are generally more difficult to design than nonrecursive ones. Traditional methods for designing recursive digital fil-

---

[3] This agrees with conclusions arrived at in [8, 7].

(p,q)	$\hat{h}(D)$	NNTE(dB)
$(7,7)$	$\frac{10^{-4}(D-D^2+3D^3-5D^4+15D^5+174D^6-196D^7)}{1-.9929D+.2163D^2-.0486D^3+.0263D^4-.0116D^5+.0099D^6-.0031D^7}$	-29.7433
$(8,7)$	$\frac{10^{-4}(D-D^2+3D^3-5D^4+15D^5+176D^6-195D^7)}{1-.9901D+0.2165D^2-.0479D^3+.0271D^4-.0111D^5+.0111D^6-.0038D^7+.006D^8}$	-31.3082
$(10,7)$	$\frac{10^{-4}(D-D^2+3D^3-5D^4+15D^5+174D^6-195D^7)}{1-.9883D+.2166D^2-0.0474D^3+0.0276D^4-.0106D^5+.0117D^6-.0034D^7+.007D^8-.0011D^9+.0046D^{10}}$	-32.5621
$(10,10)$	$\frac{10^{-4}(D-D^2+2D^3-5D^4+15D^5+176D^6-194D^7-66D^8+45D^9+23D^{10})}{1-.9593D-.1493D^2+.1746D^3+0.0709D^4-0.0193D^5+.008D^6-.0025D^7+.0045D^8-.0008D^9+.0029D^{10}}$	-34.5509

Table 4: ARMA approximations for $h(D)$ of Example 3 using Algorithm 1 and their achievable NNTE's

(p,q)	$\hat{h}_2(D)$	NNTE(dB)
$(2,2)$	$\frac{.0186-.0156D-.0046D^2}{1-.7679D-.0171D^2}$	-26.4861
$(4,1)$	$\frac{.0186-.0197D}{1-.9882D+0.2126D^2-.0453D^3+.0233D^4}$	-29.7532
$(4,4)$	$\frac{.0186-.0197D-.0095D^2+.0089D^3+.0012D^4}{1-.9852D-.299D^2+.3965D^3-.0219D^4}$	-31.6322
$(8,8)$	$\frac{10^{-4}(186-186D-45D^2+15D^3-4D^4+18D^5-18D^6+25D^7+6D^8)}{1-.9285D-.0271D^2+.0343D^3-.047D^4+.0645D^5-.1211D^6+.1158D^7+.0013D^8}$	-37.4376

Table 5: ARMA approximations for $h_2(D)$ of Example 3 using Algorithm 1 and their achievable NNTE's

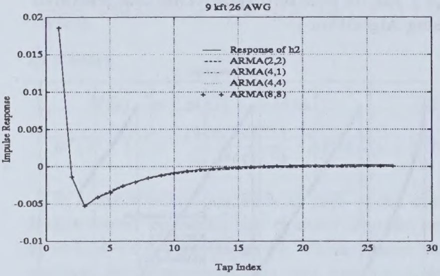


Figure 8: Pole-zero approximations of $h_2(D)$ of Example 3 generated using Algorithm 1.

ters make use of the existing wealth in analog filter design methods by applying a transformation (as in the impulse-invariant or bilinear methods) on an analog filter that meets the given desired frequency response characteristics [24, 25].

Algorithm 1 can be used as a computationally-efficient time-domain recursive digital filter design method, however, the design criterion is now related to *matching second-order statistics.*

As pointed out earlier, recursive digital filters can usually achieve a similar magnitude response to an FIR filter with much fewer parameters, especially in situations where there is a sharp transition between band edges. However, we should point out that the exact amount of reduction in implementation cost depends on the architecture of the signal processor used for filter implementation. Moreover, recursive digital filters have other limitations including their inability to have an exactly constant group delay and their sensitivity to finite-precision effects such as coefficient errors, quantization noise, overflow, and limit cycles. The finite-precision effects can be partially offset by using alternative realizations (other than the direct pole-zero form) such as parallel, cascade, or lattice forms. The latter realization can be constructed using the set of forward and backward reflection coefficients generated by Algorithm 1.

5 Conclusion

We have derived a computationally-efficient stable algorithm that reduces the implementation complexity of the MMSE-DFE by accurately approximating its long FIR feedforward and feedback filters by pole-zero filters with fewer coefficients.

Extensive simulation results on a representative loop from the HDSL environment demonstrated the viability of the proposed algorithm. Finally, a host of other applications where the derived algorithm could be used were described and discussed.

References

[1] D. Lee, B. Friedlander, and M. Morf. Recursive Ladder Algorithms for ARMA Modeling. *IEEE Transactions on Automatic Control*, AC-27(4):753–764, August 1982.

[2] J. Tu. *Theory, Design, and Application of Multi-Channel Modulation for Digital Communications.* PhD thesis, Stanford University, June 1991.

[3] C. Burrus and T. Parks. Time Domain Design of Recursive Digital Filters. *IEEE Transactions on Audio and Electroacoustics*, AU-18(2):137–141, June 1970.

[4] A. Evans and R. Fischl. Optimal Least Squares Time-Domain Synthesis of Recursive Digital Filters. *IEEE Transactions on Audio and Electroacoustics*, AU-21(1):61–65, February 1973.

[5] A. Shaw. An Optimal Method for Identification of Pole-Zero Transfer Functions. In *ISCAS Proceedings, San Diego, CA.*, pages 2409–2412, 1992.

[6] G. Long, D. Shwed, and D. Falconer. Study of a Pole-Zero Adaptice Echo Canceller. *IEEE Transactions on Circuits and Systems*, CAS-34:765–769, July 1987.

[7] G. Davidson and D. Falconer. Reduced Complexity Echo Cancellation Using Orthonormal Functions. *IEEE Transactions on Circuits and Systems*, CAS-38, January 1991.

[8] P. Crespo and M. Honig. Pole-Zero Decision Feedback Equalization with a Rapidly Converging Adaptive IIR Algorithm. *IEEE Journal on Selected Areas in Communications*, 9(6):817–829, August 1991.

[9] N. Al-Dhahir and J. Cioffi. Fast Algorithms for the Computation of the FIR MMSE-DFE. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, March 1992.

[10] N. Al-Dhahir and J. Cioffi. MMSE Decision Feedback Equalizers : Finite-Length Results. Submitted to IEEE Transactions on Information Theory, August 1993.

[11] S. Kay. *Modern Spectral Estimation: Theory and Applications.* Prentice Hall, 1988.

[12] T. Kailath. Linear Estimation for Stationary and Near-Stationary Processes. In *Lecture Notes for Arab School on Science and Technology.* Hemisphere Publishing Co., 1983. Summer School on Signal Processing.

[13] F. Ling and J. Proakis. A Generalized Multichannel Least Squares Lattice Algorithm Based on Sequential Processing Stages. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-32(2):338–343, April 1984.

[14] H. Akaike. A New Look at the Statistical Model Identification. *IEEE Transactions on Automatic Control*, AC-19(6):716–723, December 1974.

[15] J. Chow. On Estimating the Orders of an Autoregressive Moving Average Process with Uncertain Observations. *IEEE Transactions on Automatic Control*, AC-17, October 1972.

[16] S. Pillai, T. Shim, and D. Youla. A New Technique for ARMA-System Identification and Rational Approximation. *IEEE Transactions on Signal Processing*, 41(3):1281–1304, March 1993.

[17] N. Al-Dhahir and J. Cioffi. Bandwidth Optimization for Combined Equalization and Coding Transceivers with Emphasis on the MMSE-DFE. In *IEEE Milcom Conference*, October 1993.

[18] J. Salz. Optimum Mean-Square Decision Feedback Equalization. *Bell System Tech. J.*, 52(8), October 1973.

[19] J. Cioffi, G. Dudevoir, M. Eyuboglu, and G.D. Forney. Minimum Mean-Square-Error Decision Feedback Equalization and Coding - Part I: Equalization Results. *IEEE Transactions on Communications*, 1993. To Appear.

[20] N. Al-Dhahir and J. Cioffi. Mismatched MMSE Decision Feedback Equalizers. In preparation, September 1993.

[21] J. Shynk. Adaptive IIR Filtering. *IEEE ASSP Magazine*, pages 4–21, April 1989.

[22] C. Johnson. Adaptive IIR Filtering : Open Issues. *IEEE Transaction on Information Theory*, IT-30:237–250, March 1984.

[23] J. Cioffi. Channel Identification for High-Speed Voiceband Modems. Preprint, March 1991.

[24] L. Jackson. *Digital Filters and Signal Processing.* Kluwer Academic Publishers, 1986.

[25] T. Parks and C. Burrus. *Digital Filter Design.* John Wiley & Sons Inc., 1987.

Fast Algorithms for Designing Stack Filters

J. Yoo, K. L. Fong, E. J. Coyle, G. B. Adams III
School of Electrical Engineering
Purdue University
West Lafayette, IN 47907-0501

ABSTRACT

A fast, highly-parallel algorithm is developed for determining a stack filter which minimizes the mean absolute error criterion. It retains the iterative nature of the present adaptive algorithm, but significantly reduces the number of iterations required to converge to an optimal filter. It thus allows stack filters with large windows to be designed very quickly.

1 Introduction

Stack filters are a class of sliding window, nonlinear filters which satisfy the weak superposition property known as the threshold decomposition and the ordering property called the stacking property [1, 2].

An analytical technique for determining a stack filter which is optimal for estimation under the mean absolute error (MAE) criterion was developed in [3]. Unfortunately, this technique requires knowledge of the joint threshold crossing statistics of the signal and noise processes. Such knowledge is rarely avaialable. Furthermore, the complexity of the optimization procedure increases exponentially as a function of the window width of the filter.

To minimize these problems, the adaptive stack filtering algorithm in [4] were developed. Unfortunately, this adaptive algorithm does not readily lend itself to parallelization and its execution time increases faster than linearly as a function of the window size.

In this paper, a new stack filter training algorithm is developed. It uses a new method of enforcing the stacking property which allows many operations to be performed in parallel.

The new algorithm is similar to the current adaptive algorithm [4] in that it requires an image and a corrupted version of that image to train the filter. The training takes place as the level crossing statistics of these images are gathered.

2 Review of Optimal Stack Filtering Algorithms

Images will be assumed to take values on a lattice S. Each point in the lattice will be denoted by $s \in S$. A gray scale image X with pixel values ranging between 0 and M

may be represented as the sum of binary-valued images,

$$X(s) = \sum_{l=1}^{M} x_l(s),$$

where, for each l, the binary image $x_l(s)$ is obtained by thresholding $X(s)$ at level l.

$$x_l(s) = \begin{cases} 1, & X(s) \geq l \\ 0, & X(s) < l \end{cases} .$$

Let W be the size P window of the filter and and let $W(s)$ be the array of P points of the image that appears in the window W when its reference point is at position s. Then, the window array $W_X(s)$ of the image X can be similarly thresholded, so that

$$W_X(s) = \sum_{l=1}^{M} w_{X,l}(s) .$$

Each stack filter $S_f(\cdot)$ is defined by a Boolean function $f(\cdot)$ which satisfies a *stacking* property [1]; for all $k \leq l$

$$f(w_{X,k}(s)) \geq f(w_{X,l}(s)) . \tag{1}$$

A Boolean function has this property if and only if it is positive.

The following superposition property holds for the stack filter S_f based on the positive Boolean function $f(\cdot)$ with the window $w_{X,l}$:

$$S_f(W_X(s)) = S_f\left(\sum_{l=1}^{M} w_{X,l}(s)\right) = \sum_{l=1}^{M} S_f(w_{X,l}(s)) = \sum_{l=1}^{M} f(w_{X,l}(s))$$

Any Boolean function f of P variables can be represented by a length 2^P decision vector D, $D = (d_1, d_2, \cdots, d_{2^P})$, where $d_i = f(\alpha_i)$ and α_i is one of the binary sequences which correspond to the window arrays with P elements. The problem of finding an optimal Boolean function is thus equivalent to finding its decision vector.

The goal is to determine a stack filter which minimizes the mean absolute error between the output of the filter and some desired image, given a corrupted version of the desired image as input to the filter [3]. If X is the desired image, and $\tilde{X}$ is the corrupted version observed by the filter, then, as shown in [3], the error to be minimized by proper choice of f is

$$\begin{aligned} \mathrm{MAE}_f &= E\{\,|\,X(s) - S_f(W_{\tilde{X}}(s))\,|\,\} \\ &= \sum_{l=1}^{M} E\left\{|x_l(s) - f(w_{\tilde{X},l}(s))|\right\} \end{aligned} \tag{2}$$

The problem of finding a positive Boolean function which minimizes this error can be formulated as a linear program [3]:

$$\begin{gathered} \text{minimize } \textstyle\sum_{i=1}^{2^P} C_i \cdot d_i \\ \text{subject to the constraints:} \\ \\ d_i \leq d_j \text{ if } \alpha_i \leq \alpha_j \\ 0 \leq d_j \leq 1 \text{ for all } j. \end{gathered} \tag{3}$$

The number of constraints on f implied by the stacking property grows exponentially in the window size of the filter. Also knowledge of the joint statistics of the image X and the process which corrupted it is required for computing the coefficients of the cost function.

The adaptive stack filtering algorithm developed in [4] minimizes these problems. In the adaptive stack filtering procedure the desired image, X, and its noise-corrupted version $\tilde{X}$ are used to train a positive Boolean function f to minimize the error between the filter output $\hat{X}$ and the desired image X when $\tilde{X}$ is the input to the filter S_f.

In the adaptive algorithm, each entry of the decision vector represents a soft decision; each decision variable can take on values in [0,1] with resolution $1/N$, where N is some positive integer.

Let d_i represent the ith entry of the decision vector, and suppose that all of the entries of the decision vector satisfy the stacking constraints. The next step in the algorithm is the taking of an observation from the training images. Each observation consists of a binary window vector found at some position s and threshold level l of the noise corrupted version of the desired image. If this window vector is the binary vector α_i, then we update d_i either by incrementing it by $1/N$ or decrementing it by $1/N$. It is incremented if the bit at position s and threshold level l in the desired image is a 1; otherwise, it is decremented. If there were no stacking constraints, d_i would represent the relative frequency with which the filter should output a 1 versus a 0 when the vector α_i is observed in the window.

Immediately after the observation is made, a check is made to determine if the new value of d_i violates any of the stacking constraints. If it is found to violate a stacking constraint with entry d_j of the decision vector, then these two entries are swapped. This process of checking and swapping continues until all entries of the decision vector obey the stacking constraint. Another observation is then taken from the training images and the process is repeated.

At the end of training, a hard decision must be determined. For each possible binary window vector α_i the corresponding entry d_i in the decision variable is thresholded at $1/2$. If it is greater than or equal to $1/2$, then the optimal filter should put out a 1 whenever α_i is observed in the window; otherwise, it should put out a 0. In this fashion, the truth table for an optimal positive Boolean function is determined.

An optimal filter is eventually reached by this algorithm [4].

3 The New Highly-Parallel Algorithm

When the window of the filter is of size P, the window array $W(s)$ contains P pixels. The thresholded binary window array $w_l(s)$ at s therefore corresponds to exactly one of the 2^P binary vectors of length P. Let α_i, $0 \leq i \leq 2^P - 1$, denote the ith of the 2^P binary vectors of length P. Then a Boolean function $f(\cdot)$ is completely specified by the decision vector D, $D = (d_1, d_2, \cdots, d_{2^P})$, where $d_i = f(\alpha_i)$, the ith decision variable, is the output of $f(\cdot)$ when α_i is the input.

In the algorithms to be described below, each decision variable can represent a soft decision; i.e., each can take on values in [0,1] with resolution $1/N$, where N is some positive integer. In this case d_i is interpreted as the probability that the filter $f(\cdot)$ puts

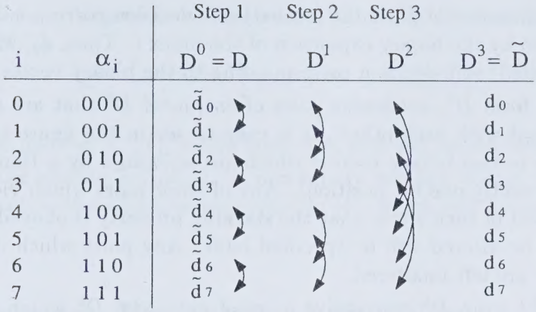

Step 1
Step 2
Step 3
i
α_i
$D^0 = D$
D^1
D^2
$D^3 = \tilde{D}$
0 000 $\tilde{d}_0$ d_0
1 001 $\tilde{d}_1$ d_1
2 010 $\tilde{d}_2$ d_2
3 011 $\tilde{d}_3$ d_3
4 100 $\tilde{d}_4$ d_4
5 101 $\tilde{d}_5$ d_5
6 110 d_6 d_6
7 111 $\tilde{d}_7$ d_7

Figure 1: Parallel scheme for enforcing the stacking property for a size 3 window.

out a 1 when α_i is the input.

For the sake of an efficient implementation, each decision variable is rescaled so that it takes on integer values in $[-N/2, N/2]$. A decision vector with strictly hard decisions can be recovered from the rescaled soft decision vector at any time by thresholding it at 0. Both algorithms begin by setting $d_i = 0$ for all i.

If the binary sequence α_i is observed at position s in a corrupted version of an image, it must be determined whether the true value of the pixel at position s in the uncorrupted image is 0 or 1. If it is 0, then d_i is decremented by 1; otherwise, d_i is incremented by 1.

Unlike the original algorithm, the stacking property will not be enforced on D after each observation by the new algorithms. Instead, it will be checked and enforced only after L observations have been taken.

Suppose that L corresponds to all possible observations from the training images, then, for each i, the number of increments performed on d_i is the cost incurred if the filter puts out a 0 for each observation of α_i. The number of decrements performed is the cost incurred if the filter puts out a 1 each time α_i is observed. Thus, if d_i is thresholded at 0 to produce b_i (i.e., $b_i = 0$ if $d_i \leq 0$, $b_i = 1$ if $d_i > 0$) the result is the optimal hard decision b_i for α_i.

Gather these hard decisions together in a binary vector $B(L)$. This is the hard decision vector for a Boolean function which minimizes the mean absolute error between its output and the desired image when the corrupted version of that image appears at the input. In other words, it is an optimal Boolean function for filtering the corrupted image used in the training process. Note, though, that $B(L)$ usually will not specify a positive Boolean function.

At this point some method for checking and enforcing the stacking property must be applied. The new algorithm given below for these tasks achieves a high degree of parallelism. It resembles a merge-sort algorithm: pairs of vector elements are compared in parallel and a local stacking property [3] is enforced at each step until all related index pairs are compared and the new decision vector satisfies the global stacking property. These operations are all applied to D, the vector of unconstrained soft decisions. Figure 1 shows how the algorithm would execute on the length 8 decision vector of a stack filter with a window of width three.

In Figure 1, first make the assignment $D^0 = D$. Recall that D is ordered in such a way that d_i, the ith entry of D, is the (scaled) soft decision corresponding to the binary vector that is given by the binary expansion of the index i. Thus, d_6, which is at position 6 in D, is the (scaled) soft decision corresponding to the binary vector $6 = 110$.

To obtain D^1 from D^0, successive pairs of entries of D^0 that are a distance $2^0 = 1$ apart are compared with each other. It is easy to see in the figure that each of these pairs corresponds to two binary vectors which are separated by a Hamming distance of 1 (they differ in exactly one bit position). Any of these pairs which violate the stacking property are altered in such a way that the stacking property is obeyed. The constraints on how they can be altered will be specified later. Any pairs which do not violate the stacking property are left unaltered.

To obtain D^{k+1} from D^k, successive pairs of entries of D^k which are a distance 2^k from each other are compared with each other. Note that each of these pairs correspond to two binary vectors which are separated by a Hamming distance of 1 (they differ in exactly one bit position) because of the ordering of the entries. Any pairs which violate the stacking property are altered so that they obey the stacking property. Any pairs which do not violate the stacking property are left unaltered.

This process continues until $k = P$, where P is the window size of the filter. At this point, the output for every binary vector will have been compared to the outputs for all binary vectors which are a Hamming distance of 1 away, and every locally observed violation of the stacking property would have been corrected. Thus, all entries of the vector obey the local stacking property defined in [3]. This is necessary and sufficient for the subdecision vector D^P to satisfy the global stacking property [3]. Thus, if D^P were thresholded at zero, the result would be a (hard) decision vector corresponding to a positive Boolean function.

It is now necessary to specify how the stacking constraint is enforced on pairs which violate the stacking property. As above, $D^k(n)$ is the subdecision vector of $D(n)$ produced at the kth stage of the algorithm, and ${d_i}^k$ is the ith entry of $D^k(n)$. The index n implies that the decision vector has been updated n times; it will usually be omitted.

Theorem 3.1 *If at each stage of the algorithm described above, each pair* ${d_i}^k$ *and* ${d_j}^k$ *found to violate the stacking property is used to define* ${d_i}^{k+1}$ *and* ${d_j}^{k+1}$ *such that*

$$ {d_j}^k \le {d_i}^{k+1} \le {d_j}^{k+1} \le {d_i}^k, $$

then the vector D^P *will satisfy the global stacking property.*

A method for determining the new values of local pairs which violate the stacking property is now introduced. It seeks to preserve the statistical properties of the unconstrained soft decision vector, $D(\cdot)$.

The method seeks to set each violating pair of decision variables to their statistical average, without resorting to floating point data types and without incurring any ill behavior that might be associated with roundoff errors. This constrains the algorithm to integer data types, so it satisfies the condition of the rescaled soft decision vector mentioned earlier. Our solution is to use the ceiling and floor operators on the average of the decision variable pair. In other words, for any decision variable pair which violates the local stacking property after the kth step,

IF ${d_i}^k > {d_j}^k$ and $i < j$, where the observed binary sequences α_i and α_j are Hamming distance 1 apart

THEN,

$$d_i^{k+1} = \left\lfloor \frac{d_i^k + d_j^k}{2} \right\rfloor \text{ and } d_j^{k+1} = \left\lceil \frac{d_i^k + d_j^k}{2} \right\rceil ,$$

where for any real number x, $\lfloor x \rfloor$, is the floor of x, and $\lceil x \rceil$, is the ceiling of x,

Note that if the update period, L, is one binary window vector update, the above method is identical to the adaptive algorithm in [4], which will hereafter be called the LSC algorithm.

4 Convergence Property of the New Algorithm

We now show that the new algorithm converges to an optimal positive Boolean function. To this end, let $D(n)$ again denote a soft decision vector which does not have the stacking property among its variables. Let $\tilde{D}(n)$ denote a soft decision vector with the stacking property, where n denotes the number of observations of the training data. In the rest of this paper, "decision vector" refers to a soft decision vector satisfying the stacking constraints. We will use "unconstrained decision vector" for a decision vector which does not necessarily satisfy the stacking constraints.

After they are thresholded to produce hard decisions, $D(n)$ and $\tilde{D}(n)$ correspond to an arbitrary Boolean function and a positive Boolean function, respectively. f_{opt} represents the optimal positive Boolean function to be reached. For a comparison of convergence rates, we use the notation $X(n)$ for the decision vector in the LSC algorithm which satisfies the stacking constraint [4].

In general, the convergence behavior of the new algorithm is similar to that of the LSC algorithm. The primary difference is when and how the stacking constraint is enforced. By showing that both the average local drift and the average global drift of the decision vector in the new algorithm is toward the optimal positive Boolean function, we will prove the convergence of the new algorithm.

The following theorem from [4] says that the unconstrained decision vector, on average, drifts closer to the optimal positive Boolean function over a training period of length L if the number of steps, N, is large enough.

Theorem 4.1 *(see Theorem 5.4 in [4])*

$$E[||D(n+L) - f_{opt}|| \mid F_n] \leq ||\tilde{D}(n) - f_{opt}|| - \rho + \frac{L^2}{2||\tilde{D}(n) - f_{opt}||} \qquad (4)$$

where ρ is a constant which depends linearly on the number of steps N.
□

If the stacking constraint is not imposed, the new adaptive algorithm is the same as the LSC algorithm. We thus use this theorem without going into any details about its proof. If N is large enough so that ρ is greater than the last term of Equation (4), then Theorem 4.1 says that when the stacking constraint is not imposed, the decision vector will, on average, move closer to the optimal positive Boolean function over a period of length L.

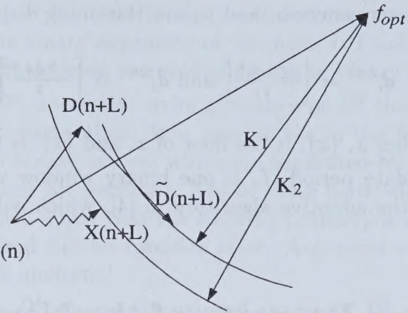


Figure 2: Comparison of the local drift property of the new algorithm to that of the LSC algorithm.

Now what happens when we add the stacking property? We need to show that the new algorithm does not move the decision vector away from the optimal decision vector over a period of length L even after the stacking constraints are enforced. Further, we show that the decision vector in the new algorithm moves faster to an optimal decision vector than the decision vector in the LSC algorithm. Figure 2 illustrates this point. In other words, the bound of the distance of the decision vector from the optimal decision vector, $K1$, in the new algorithm is always less than the bound $K2$ obtained by the LSC method.

Let $D^k(n)$ again be a subdecision vector of $D(n)$ after the kth step. Then, there are a total of $P+1$ subdecision vectors of $D(n)$ with $D^0(n) = D(n)$ and $D^P(n) = \tilde{D}(n)$ for the window width P. In the following Lemma, we prove that the distance between the decision vector $\tilde{D}(n+L)$ and the unconstrained decision vector $D(n+L)$ is bounded.

Lemma 4.1

$$||\tilde{D}(n+L) - D(n+L)||^2 \leq (3L^2+1)/2 \tag{5}$$

where $||\cdot||$ *denotes the Euclidean norm.*

Proof: If $L = 1$, adding the stacking condition is just a simple swap operation, so (5) clearly holds. If $L \geq 2$, first suppose that there is only one stacking violation between the ith and jth decision variables in the unconstrained decision vector $D(n+L)$, with $i < j$. Then,

$$0 < d_i(n+L) - d_j(n+L) \leq L \tag{6}$$

where $d_i(n+L)$ and $d_j(n+L)$ are the ith and jth decision variables of the unconstrained decision vector $D(n+L)$. If there is no stacking violation in the succeeding subdecision vectors $D^k(n+L)$ for $1 \leq k \leq P$, then

$$\begin{aligned} ||\tilde{D}(n+L) - D(n+L)||^2 &= (\tilde{d}_i - d_i)^2 + (\tilde{d}_j - d_j)^2 \\ &= \left(d_i - \left\lfloor \frac{(d_i+d_j)}{2} \right\rfloor\right)^2 + \left(d_j - \left\lceil \frac{(d_i+d_j)}{2} \right\rceil\right)^2 \\ &= 2\left(d_i - \left\lfloor \frac{(d_i+d_j)}{2} \right\rfloor\right)^2 \end{aligned}$$

$$\begin{aligned} &= 2\left\lceil\frac{(d_i - d_j)}{2}\right\rceil^2, \\ &\le 2\left\lceil\frac{L}{2}\right\rceil^2 \qquad (\text{ from } (6)) \\ &\le L^2, \end{aligned} \tag{7}$$

where the last inequality holds since $L \geq 2$. Therefore, (5) clearly holds. In the above expressions, we omitted the index $n + L$ for simplicity.

If there are more stacking violations in the succeeding decision vectors, then the bound can be found as follows. We again use the fact that $L^2 \geq a^2 + b^2$ where $L = |a| + |b|$. Suppose we have a maximum number of stacking violations in the succeeding decision vectors, then

$$\begin{aligned} ||\tilde{D}(n+L) - D(n+L)||^2 &= L^2 + \overbrace{1 + 1 + \cdots + 1 + 1}^{\left\lfloor\frac{(d_i(n+L) - d_j(n+L))}{2}\right\rfloor} + \underbrace{1 + 1 + \cdots + 1 + 1}_{\left\lceil\frac{(d_i(n+L) - d_j(n+L))}{2}\right\rceil} \\ &\le L^2 + \left\lfloor\frac{d_i(n+L) - d_j(n+L)}{2}\right\rfloor^2 + \left\lceil\frac{d_i(n+L) - d_j(n+L)}{2}\right\rceil^2 \\ &\le L^2 + \left\lfloor\frac{L}{2}\right\rfloor^2 + \left\lceil\frac{L}{2}\right\rceil^2 \\ &\le (3L^2 + 1)/2 \end{aligned} \tag{8}$$

where the maximum occurs when L is an odd number.

Now suppose there are an arbitrary number of stacking violations in $D(n + L)$. Since $L^2 \geq a^2 + b^2$ where $L = |a| + |b|$, (7) and (8) still hold.
□

The following lemma shows that the inner product between $\tilde{D}(n + L) - D(n + L)$ and f_{opt} is always greater than zero.

Lemma 4.2

$$(\tilde{D}(n + L) - D(n + L)) \cdot f_{opt} \geq 0 \tag{9}$$

where $\cdot$ denotes a vector product.

Proof: We can rewrite (9) by using the subdecision vectors as follows:

$$(\tilde{D}(n + L) - D(n + L)) \cdot f_{opt} = \sum_{l=0}^{P-1} (D^{P-l}(n + L) - D^{P-l-1}(n + L)) \cdot f_{opt} \tag{10}$$

where P is, as always, the size of the filter window.

Suppose that there are M local stacking violations in the $(P - l - 1)$th decision vector and $i(m) < j(m)$ where $i(m)$ and $j(m)$ indicate the decision variable pairs which have the violation. Then,

$$D^{P-l}(n + L) - D^{P-l-1}(n + L) = \sum_{m=1}^{M} A_m(\mathbf{1}_{j(m)} - \mathbf{1}_{i(m)})$$

where $A_m = d_{i(m)}^{P-l-1}(n+L) - \left\lfloor \frac{(d_{i(m)}^{P-l-1}(n+L)+d_{j(m)}^{P-l-1}(n+L))}{2} \right\rfloor$, $A_m \geq 0$ and $\mathbf{1}_p$ is a length 2^P vector whose entries are all 0's except the pth entry which is 1.

But, for any positive Boolean function f, and any pair of $i(m)$, $j(m)$ with $i(m) < j(m)$,

$$A_m(\mathbf{1}_{j(m)} - \mathbf{1}_{i(m)}) \cdot f \geq 0,$$

Therefore, $(D^{P-l}(n+L) - D^{P-l-1}(n+L)) \cdot f_{opt} \geq 0$ for all l and

$$(\tilde{D}(n+L) - D(n+L)) \cdot f_{opt} \geq 0 \tag{11}$$

□

The following lemma shows that the inner product between $\tilde{D}(n+L) - D(n+L)$ and $D(n+L)$ is always less than zero.

Lemma 4.3

$$(\tilde{D}(n+L) - D(n+L)) \cdot D(n+L) \leq 0 \tag{12}$$

where · denotes a vector product.

Proof: To prove this lemma, we need to introduce the following notation. Let the set of decision variables which are incremented or decremented after the global stacking condition is added, be denoted by Q^+ and Q^-, respectively. Then, it is clear that

$$\sum_{i \in Q^+} (\tilde{d}_i(n+L) - d_i(n+L)) + \sum_{j \in Q^-} (\tilde{d}_j(n+L) - d_j(n+L)) = 0.$$

In other words, the number of increments always equals the number of decrements over a training period of length L after the global stacking condition is enforced. However, if an increment happens, the corresponding decrement always happens associated with a larger value of unconstrained decision variable than an increment. Therefore,

$$\sum_{i \in Q^+} (\tilde{d}_i(n+L) - d_i(n+L))d_i(n+L) + \sum_{j \in Q^-} (\tilde{d}_j(n+L) - d_j(n+L))d_j(n+L) \leq 0.$$

and,

$$\begin{aligned}(\tilde{D}(n+L) - D(n+L)) \cdot D(n+L) &= \sum_{i \in Q^+} (\tilde{d}_i(n+L) - d_i(n+L))d_i(n+L) \\ &\quad + \sum_{j \in Q^-} (\tilde{d}_j(n+L) - d_j(n+L))d_j(n+L) \\ &\leq 0\end{aligned}$$

□

Using the above three lemmas, we now can prove that the unconstrained decision vector $D(n+L)$ in the new algorithm always moves faster to the optimal positive Boolean function over a training period of length L than in the LSC algorithm in [4].

Theorem 4.2 *If* $||\tilde{D}(n) - f_{opt}|| - L > 0$,

$$||\tilde{D}(n+L) - f_{opt}|| \leq ||D(n+L) - f_{opt}|| + \frac{(2M)}{||\tilde{D}(n) - f_{opt}|| - L} \tag{13}$$

where $M = (\frac{3L^2+1}{4})$.

Proof: If we take the square on the left term of (13), then

$$\begin{aligned}||\tilde{D}(n+L)-f_{opt}||^2 &= ||\tilde{D}(n+L)-D(n+L)||^2+||D(n+L)-f_{opt}||^2\\ &+2(\tilde{D}(n+L)-D(n+L))\cdot(D(n+L)-f_{opt}).\end{aligned}$$

By Lemma 4.1, Lemma 4.2, Lemma 4.3, and the fact that $\sqrt{a^2+b} \leq a+\frac{b}{2a}$ if $a>0$.

$$||\tilde{D}(n+L)-f_{opt}|| \leq ||D(n+L)-f_{opt}||+\frac{(2M)}{||D(n+L)-f_{opt}||}$$

where $M=(\frac{3L^2+1}{4})$.

By using the triangle inequality and the the following fact from [4]

$$||D(n+L)-\tilde{D}(n)|| \leq L,$$

we obtain

$$||\tilde{D}(n+L)-f_{opt}|| \leq ||D(n+L)-f_{opt}||+\frac{(2M)}{||\tilde{D}(n)-f_{opt}||-L}. \tag{14}$$

where $M=(\frac{3L^2+1}{4})$.
□

If $||\tilde{D}(n)-f_{opt}||$ is large enough – this is true when we have large N – then the last term in (13) will be small enough. Therefore, enforcing the stacking constraints does not move the decision vector away from an optimal positive Boolean function. Also, note that the bound in (14) is always less than the bound obtained for the LSC algorithm. In the LSC algorithm, the M in (14) is given by L^2. This implies that a decision vector moves to the optimal positive Boolean function faster than the decision vector in the LSC algorithm over a period of length L, as shown in Figure 2 if $L \geq 2$.

By using the above lemmas and theorems, we can also show that, in the long term, the drift of the decision vector, $\tilde{D}(n)$ is toward the optimal positive Boolean function. Since it is very similar to the LSC approach except for some different values appearing in the bounds used in inequalities, we will not repeat the details of the proof. They can be found in Theorem 5.8 and Theorem 5.9 of [4].

References

[1] P. D. Wendt, E. J. Coyle, and N. C. Gallagher, Jr., "Stack filters," *IEEE Trans. Acoust. Speech, Signal Processing,* vol. 34, pp.898-911, Aug. 1986.

[2] J. P. Fitch, E. J. Coyle, and N. C. Gallagher, Jr., "Threshold decomposition of multidimensional rank order operators," *IEEE Trans. Circuits Syst.*, vol. 32, pp. 445-450, May 1985.

[3] E. J. Coyle and J.-H. Lin, "Stack filters and the mean absolute error criterion," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. 36, pp. 1244-1254, Aug. 1988.

[4] J.-H. Lin, T. M. Sellke, and E. J. Coyle, "Adaptive stack filtering under the mean absolute error criterion," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. 38, pp. 938-954, June 1990.

An Algebraic Approach To Null-Pole Coupling Operators

Marek Rakowski
Bostwick F. Wyman

Department of Mathematics
The Ohio State University
231 West 18th Avenue
Columbus, OH 43210
rakowski@math.ohio-state.edu
wyman@math.ohio-state.edu

Abstract

Some aspects of the operator-theoretic and module-theoretic approaches to poles and zeros of linear systems are compared, and a new algebraic definition for the null-pole coupling operator is proposed.

1 Introduction

In this expository paper, we discuss two distinct approaches to the foundations of linear systems theory: the module theoretic approach based on pole and zero modules as in [WSCP1-2] and the operator-theoretic approach based on null and pole pairs as in [BGR1-3, BR1-2]. The main thrust of the discussion is a comparison of pole and zero modules with pole pairs and null pairs, and in particular a new algebraic definition is given for the null-pole coupling operator.

The discussion here is limited to poles and zeros in the entire finite complex plane, rather than poles and zeros in more general regions. This more general case, common in the analytic approach, can be treated algebraically using localization of rings and modules (compare [WCP]). Precise local statements, as well as all proofs, will be postponed to later publications.

2 Pole Dynamics and Pole Pairs

Suppose given a transfer function $G(z) : U(z) \to Y(z)$ with $U \cong \mathbf{C}^m$, $Y \cong \mathbf{C}^p$. Following Kalman, we define the *minimal state space* or *pole module* associated with $G(z)$ by

$$P(G) = \frac{U[z]}{G^{-1}(Y[z]) \cap U[z]}.$$

The module $P(G)$ is a finite-dimensional complex vector space whose dimension n_π is called the *McMillan degree* of $G(z)$. The $\mathbf{C}[z]$ module action on $P(G)$ defines the classical dynamics action

$$A_\pi : P(G) \to P(G)$$

$$A_\pi(x) = z \cdot x$$

We use the subscript π to denote classical or pole dynamics to distinguish them from the zero dynamics which will appear in the next section.

The Kalman realization theory gives a restricted input-output mapping $G^\# : U[z] \to Y(z)/Y[z]$, defined by

$$G^\# = \widetilde{C_\pi} \circ \widetilde{B_\pi},$$

where the three maps $G^\#, \widetilde{B_\pi}, \widetilde{C_\pi}$ are $\mathbf{C}[z]$ module mappings given by

$$G^\#(u(z)) = G(z)u(z) \text{mod } Y[z]$$

$$\widetilde{B_\pi}(u(z)) = u(z) \text{mod } U[z] \cap G^{-1}(Y[z])$$

$$\widetilde{C_\pi}(x) = G^\#(u(z)) \text{ if } \widetilde{B_\pi}(u(z)) = x.$$

In the usual discrete-time intuition, $G^\#$ gives the future ouput of a past input string, ending at time $t = 0$, $\widetilde{B_\pi}$ is the reachability map which maps a past input to the resulting state at time $t = 1$ and $\widetilde{C_\pi}$ is the observability map giving the future output of a state. We can define also the instantaneous mapping $B_\pi : U \to P(G)$ and $C_\pi : P(G) \to Y$ by

$$B_\pi(u) = \widetilde{B_\pi}(u) \text{ for } u \in U \subset U[z]$$

$$\widetilde{C_\pi}(x) = C_\pi(x) \cdot z^{-1} + \cdots \text{ mod } Y[z].$$

These constructions guarantee that

$$\pi_- G(z) = C_\pi(zI - A)^{-1} B_\pi$$

where π_- denotes the strictly proper part of the transfer function $G(z)$.

Another approach to pole dynamics is given by the idea of *pole pairs* in the analytic theory. A function $\psi \in U(z)$ is a *right pole function* for a $G(z)$ at a point $\lambda \in \mathbf{C}$ if

(i) ψ is analytic and nonzero at λ,

(ii) $\psi(z) = (z - \lambda)^\kappa G(z)\phi(z)$ for some positive integer κ and a function $\phi \in U(z)$ analytic at λ.

The largest integer κ for which (ii) can happen is called the *order* of ψ as a right pole function for G at λ.

A collection $\{\psi_1, \psi_2, ..., \psi_M\}$ of right pole functions for G at λ of orders $\kappa_1, \kappa_2, ..., \kappa_M$ is said to be *canonical* if

(i) $\psi_1(\lambda), \psi_2(\lambda), ..., \psi_M(\lambda)$ are linearly independent,

(ii) The sum of $\sum_{i=1}^{M} \kappa_i$ is maximal.

Let $\{\psi_1, \psi_2, ..., \psi_M\}$ be a canonical set of right pole functions for G at λ of orders $\kappa_1, \kappa_2, ..., \kappa_M$. The information about the pole of G at λ contained in $\{\psi_1, \psi_2, ..., \psi_M\}$ can be encoded in a pair of matrices as follows. If $\psi_i(z) = \sum_{j=1}^{\infty}(z-\lambda)^{i-1}\psi_{i,j}$, let

$$C_\lambda = \begin{bmatrix} \psi_{1,1} & \psi_{1,2} & \cdots & \psi_{1,\kappa_1} & \cdots & \psi_{M,1} & \psi_{M,2} & \cdots & \psi_{M,\kappa_M} \end{bmatrix}$$

and let

$$A_\lambda = \begin{bmatrix} J_{\kappa_1}(\lambda) & & \\ & \ddots & \\ & & J_{\kappa_1}(\lambda) \end{bmatrix}$$

with $J_{\kappa_i}(\lambda)$ a $\kappa_i \times \kappa_i$ Jordan cell with the eigenvalue λ. If S is a nonsingular matrix of the same size as A, the pair $(CS, S^{-1}AS)$ is called a *right pole pair* for *G at* λ.

Let $\lambda_1, \lambda_2, ..., \lambda_p$ be the poles of G . For each λ_i, pick a right pole pair (C_i, A_i) for G at λ_i. Let

$$C_\pi = \begin{bmatrix} C_1 & \dots & C_p \end{bmatrix}$$

and let

$$A_\pi = \begin{bmatrix} A_1 & & \\ & \ddots & \\ & & A_p \end{bmatrix}.$$

The close connection between these two theories is based on the important fact that if (C_π, A_π, B_π) are the mappings discussed above associated with a Kalman realization, then (C_π, A_π) is a pole pair. Furthermore all pole pairs are similar to a pole pair associated with a realization. We omit the tedious verification, which involves the fundamental structure results for modules over principal ideal domains.

3 Zero Dynamics and Null Pairs

Another basic object of the module-theoretic approach to systems is the zero module, introduced originally by Wyman and Sain [WS1]. If $G(z)$ is a transfer function, its *zero module* in the finite plane is the C$[z]$-module

$$Z(G) = \frac{G^{-1}(Y[z]) + U[z]}{\ker G + U[z]}.$$

The elements of $Z(G)$ are inputs which give polynomial outputs, modulo inputs which are polynomial or which give identically zero outputs. The action of z on $Z(G)$ gives a *zero-dynamics map* $A_\zeta : Z(G) \rightarrow Z(G)$. An alternative "output side " description ([SWP, BRW]) will be useful. Define

$$Z_{out}(G) = \frac{G(z)U(z) \cap Y[z]}{G(z)U[z] \cap Y[z]}.$$

We can check that $Z(G)$ and $Z_{out}(G)$ are isomorphic $\mathbf{C}[z]$-modules, suing the map $u(z) \mapsto G(z)u(z)$, and we will frequently identify $Z(G)$ with $Z_{out}(G)$.

Our next goal is to construct a Kalman input map for zero dynamics, $\widetilde{B_\zeta} : Y[z] \to Z(G)$. For motivation, assume temporarily that $G(z)$ maps $U(z)$ *onto* $Y(z)$. Then

$$Z_{out}(G) = \frac{Y[z]}{G(z)U[z] \cap Y[z]}$$

and we can define $\widetilde{B_\zeta}$ as the natural projection.

In general, $G(z)U(z) \subset Y(z)$ is a proper subspace, and $H = G(z)U(z) \cap Y[z]$ is a free submodule of $Y[z]$. Since $Y[z]/H$ is torsion-free and finitely generated, it is also free. Therefore H has a free complement in $Y[z]$: there exist (many possible) free submodules F such that

$$Y[z] = G(z)U(z) \cap Y[z] \oplus F.$$

Choice of any such F defines a projection, or surjective $\mathbf{C}[z]$-map

$$\widetilde{B_\zeta} : Y[z] \to \frac{G(z)U(z) \cap Y[z]}{G(z)U[z] \cap Y[z]}$$

We emphasize that the F and the corresponding $\widetilde{B_\zeta}$ are not unique, but we make a fixed choice for the whole discussion. Varying choices are closely related with the theory of generalized inverses, discussed extensively in [BRW]. We think of the chosen couple (A_ζ, B_ζ) as a controllable input-state pair.

We recall now the definition of a null pair of the transfer function $G(z)$ .Null pairs were introduced originally for matrix polynomials [Ro, GLR] and later for regular rational matrix functions (see [BGR3], [BR1].)

A function $\phi \in Y(z)$ is a *left null function* for $G(z)$ at λ of order κ if

(i) ϕ is analytic and nonzero at λ,
(ii) there exists no $\psi \in U(z)$ such that $\psi G = 0$ and $\psi(\lambda) = \phi(\lambda)$,
(iii) ϕG vanishes at λ to the order κ.

A collection $\{\phi_1, \phi_2, ..., \phi_M\}$ of left null functions for G at λ of orders $\kappa_1, \kappa_2, ..., \kappa_M$ is said to be *canonical* if

(i) $\phi_1(\lambda), \phi_2(\lambda), ..., \phi_M(\lambda)$ are linearly independent,
(ii) there is no $\psi \in Y(z)$ with $\psi G = 0$ and $\psi(\lambda) \in \text{span}\{\phi_1(\lambda), \phi_2(\lambda), ..., \phi_M(\lambda)\}$,
(iii) $\sum_{i=1}^{M} \kappa_i$ is maximal.

Let $\{\phi_1, \phi_2, ... \phi_M\}$ be a canonical set of left null functions for G at λ, of orders

$\kappa_1, \kappa_2, ..., \kappa_M$ respectively. If $\phi_i(z) = \sum_{j=1}^{\infty} \phi_{i,j}(z-\lambda)^{j-1}$, let

$$A = \begin{bmatrix} J_{\kappa_M}(\lambda) & & \\ & \ddots & \\ & & J_{\kappa_1}(\lambda) \end{bmatrix}, \quad B = \begin{bmatrix} \phi_{M,\kappa_M} \\ \vdots \\ \phi_{M,1} \\ \vdots \\ \phi_{1,\kappa_1} \\ \vdots \\ \phi_{1,1} \end{bmatrix}.$$

Suppose $\lambda_1, \lambda_2, ..., \lambda_N$ are the zeros of G in σ. For each λ_i pick a left null pair (A_i, B_i), and let

$$A_\zeta = \begin{bmatrix} A_1 & & \\ & \ddots & \\ & & A_N \end{bmatrix}, \quad B_\zeta = \begin{bmatrix} B_1 \\ \vdots \\ B_N \end{bmatrix}.$$

If S is a nonsingular matrix of the same size as A_ζ, the pair $(SA_\zeta S^{-1}, SB_\zeta)$ is called a *null pair for G* .

Just as in the case of poles, there is a close connection between the module theory of zero dynamics and the null-pair approach. In particular, if (A_ζ, B_ζ) is a state-input pair for zero dynamics as constructed above, then (A_ζ, B_ζ) defines a null-pair, and every null-pair is similar to one constructed in this way. Again we omit the tedious , but important, verifications.

4 Wedderburn-Forney spaces

If $U = \mathbf{C}^m$, $U(z) = U \otimes \mathbf{C}(z)$ has a structure of a vector space over $\mathbf{C}(z)$, and any basis for a $\mathbf{C}$-vector space U is a basis for a $\mathbf{C}(z)$-vector space $U(z)$. Thus, we can find a basis for $U(z)$ consisting of vector polynomials the sum of whose degrees equals 0. It has been observed in [Fo] that this may no longer be true if we take a subspace of $U(z)$. Let $V(z)$ be a subspace of $U(z)$, and pick a basis for $V(z)$ consisting of elements in $U[z]$ the sum of whose degrees is minimal. The degrees of vector polynomials in the basis are unique up to a permutation and the sum of these degrees is called the *defect* of the subspace.

The concept of the defect of $V(z)$ can be made precise by means of the following construction [WSCP1, WSCP2]. Denote by $\mathcal{O}_\infty$ the local ring of proper rational functions, and let $\Omega_\infty U = U \otimes \mathcal{O}_\infty$, so that $U(z) = U[z] \oplus z^{-1}\Omega_\infty U$. Denote by π_+ the projection of $U(z)$ onto $U[z]$ along $z^{-1}\Omega_\infty U$, and let π_- be the complementary projection. The *Wedderburn-Forney space* associated with $V(z)$ is the k-vector space [WSCP1]

$$\mathcal{W}(V) = \frac{\pi_-(V)}{V \cap z^{-1}\Omega_\infty U}.$$

It is easy to show that the dimension of a Wedderburn-Forney space associated with V is equal to the defect of V. The individual column degrees are closely associated with certain filtrations of $\mathcal{W}(V)$ ([GW]).

The Wedderburn-Forney spaces have been used in [WSCP1] to relate the pole and zero modules of a transfer function G. Although we will not discuss them in this paper, it is possible to define pole and zero modules at infinity for a transfer function $G(z)$, (see [CP, WSCP]), denoted by $P_\infty(G)$ and $Z_\infty(G)$. Using these modules, we can define global pole and zero modules by $\mathcal{P}(G) = P(G) \oplus P_\infty(G)$ and $\mathcal{Z}(G) = Z(G) \oplus Z_\infty(G)$.

The main result in [WSCP1] is the Fundamental Pole Zero Exact Sequence. It was shown there that we can identify $\mathcal{W}(\ker G)$ with a subspace of $\mathcal{P}(G)$, and that the following sequence of vector spaces and vector space homomorphisms is exact:

$$0 \longrightarrow \mathcal{Z}(G) \xrightarrow{\alpha} \frac{\mathcal{P}(G)}{\mathcal{W}(\ker G)} \xrightarrow{\beta} \mathcal{W}(\operatorname{im} G) \longrightarrow 0 .$$

In this paper we assume that $G(z)$ is proper, so that it has no pole at infinity and $\mathcal{P}(G) = P(G)$. Furthermore, we can restrict the map α to the finite zero module, so that the fundamental exact sequence yields

$$0 \longrightarrow Z(G) \xrightarrow{\alpha} \frac{P(G)}{\mathcal{W}(\ker G)} .$$

This mapping was considered earlier in [WS2] , where the image of α was identified with the geometric control space V^*. Furthermore, we can identify $\mathcal{W}(\ker G)$ with the corresponding R^*, but the geometric theory will not play any part in the present exposition.

5 Coupling Operators

Consider a transfer function $G(z)$ and consider the $\mathbf{C}[z]$-module $G(U[z])$. Pick a pole pair (C_π, A_π) and a null pair (A_ζ, B_ζ) for G . If G is regular, the formula for $G(U[z])$ in terms of (C_π, A_π) and (A_ζ, B_ζ) was obtained in [BGR2]. In the general case, this formula has the form [BR2]

$$\begin{aligned} GU[z] = G(z)U(z) \cap \{C_\pi(z - A_\pi)^{-1}x + h(z) :\ & x \in \mathbf{C}^{n_\pi},\ h(z) \in Y[z] \\ \text{and } & \sum_{z_0 \in \mathbf{C}} \operatorname{Res}_{z=z_0}(z - A_\zeta)^{-1} B_\zeta h(z) = \Gamma x\} \end{aligned} \qquad (*)$$

where Γ is a matrix uniquely determined by (C_π, A_π) and (A_ζ, B_ζ). The matrix Γ is called a *null-pole coupling matrix* or a *null-pole coupling operator.* The triple $((C_\pi, A_\pi), (A_\zeta, B_\zeta), \Gamma)$ is called a *spectral triple* for G.

Given an observable pair (C_π, A_π) and a controllable pair (A_ζ, B_ζ) and a matrix Γ, one can define a set S to be the right hand side of equality (*). It turns out that S is an $\mathbf{C}[z]$-module if and only if Γ satisfies the Sylvester equation [BGR2, BR2]

$$\Gamma A_\pi - A_\zeta \Gamma = B_\zeta C_\pi. \qquad (**)$$

Thus, the null-pole coupling operator necessarily satisfies the Sylvester equation. If A_π and A_ζ have disjoint spectra, the matrix Γ can be computed as the unique solution of equation (**). Otherwise equation (**) may have many solutions, and for each solution Γ there exists a rational matrix function with a spectral triple $((C_\pi, A_\pi), (A_\zeta, B_\zeta), \Gamma)$. Moreover, different solutions Γ result in different spaces S.

In the rest of this section, we carry out the main task of the paper: a new algebraic definition of the null-pole coupling operator $\Gamma : P(G) \to Z(G)$. Think of

$$P(G) = \frac{U[z]}{G^{-1}(Y[z] \cap U[z]}, \quad Z(G) = \frac{Y[z] \cap G(z)U(z)}{Y[z] \cap G(z)U[z]}.$$

The central ingredient of the construction is the map $U[z] \to Y[z]$ defined by $u(z) \mapsto \pi_+ G(z)u(z)$. Choosing a state-input pair (A_ζ, B_ζ) with $\widetilde{B_\zeta} : Y[z] \to Z(G)$, as in Section 3, we can map $U[z] \to Z(G)$ by $u(z) \mapsto \widetilde{B_\zeta}\pi_+ G(z)u(z)$. Next we check that if $G(z)u(z) \in Y[z]$ then $u(z) \mapsto 0$ in $Z(G)$, so we get a $\mathbf{C}$-linear map $\Gamma : P(G) \to Z(G)$ as promised.

Claim: The map $\Gamma : P(G) \to Z(G)$ described above is a null-pole coupling operator for $G(z)$.

We support this claim by a series of assertions, giving here only the barest ideas of the proofs.

(1) The equation (*) above holds for the map $\Gamma : P(G) \to Z(G)$.

Idea of Proof. Let $u(z) \in U[z]$, and write

$$G(z)u(z) = \pi_+ G(z)u(z) + \pi_- G(z)u(z).$$

The strictly proper part $\pi_- G(z)u(z)$ is essentially an output $\widetilde{C_\pi} x = C_\pi (z - A_\pi)^{-1} x$ for $x = \widetilde{B_\pi} u(z)$. The polynomial part $\pi_+ G(z)u(z)$ can be used for the polynomial $h(z)$ in Equation (*) . Then $\Gamma x = \widetilde{B_\zeta} h(z)$ and we only have to verify that

$$\widetilde{B_\zeta} h(z) = \sum_{z_0 \in \mathbf{C}} \operatorname{Res}_{z=z_0} (z - A_\zeta)^{-1} B_\zeta h(z)$$

This follows from (2) below.

(2) Suppose given a finite dimensional $\mathbf{C}[z]$ module X with $z \cdot x = Ax$ for $x \in X$. Then $z - A : X(z) \to X(z)$ is an invertible $\mathbf{C}(z)$–linear transformation. Let $p_A : X[z] \to X$ be the module projection defined by $x \cdot z^i \mapsto A^i x$ Then for every $x(z)$ in $X[z]$

$$p_A(x) = \sum_{z_0 \in \mathbf{C}} \operatorname{Res}_{z=z_0} (z - A)^{-1} x(z)$$

Idea of Proof. Use $(z-A)^{-1} = I + Az^{-1} + \cdots$, and the residue formula

$$\sum_{z_0 \in \mathbf{C}} \mathrm{Res}_{z=z_0}\omega = -\sum_{z_0=\infty} \mathrm{Res}_{z=z_0}\omega$$

for differential forms on the field $\mathbf{C}(z)$. Compare the appendix to [WCP]. A precise local algebraic form of this crucial result will take more work.

(3) The map Γ satisfies the Sylvester identity

$$\Gamma A_\pi - A_\zeta \Gamma = B_\zeta C_\pi .$$

Idea of Proof. Both A_π and A_ζ are given by multiplication by z in appropriate spaces. The ideas are reminiscent of the invariant subspace calculations in [WS2] .

These three items show that Γ is a null-pole coupling operator. In addition, we have two new algebraic results.

(4) The map $\Gamma : P(G) \to Z(G)$ kills $\mathcal{W}(\ker G(z))$ in $P(G)$.
Idea of Proof. Straightforward from the definitions in [WSCP1] .

(5) The map

$$\gamma : \frac{P(G)}{\mathcal{W}(\ker G(z))} \to Z(G)$$

induced because of (4) is a left inverse for the mapping α discussed above in Section 4 , where

$$\alpha : Z(G) \to \frac{P(G)}{\mathcal{W}(\ker G(z))}.$$

Idea of Proof. Not so hard. Essentially contained in [BRW] .

References

[BCR] J. A. Ball, N. Cohen and L. Rodman, Zero Data and Interpolation Problems for Rectangular Matrix Polynomials, *Linear and Multilinear Algebra*, vol. 29, pp. 53-78, 1991.

[BGR1] J. A. Ball, I. Gohberg and L. Rodman, Realization and Interpolation of Rational Matrix Functions, in *Topics in Interpolation Theory of Rational Matrix Functions* (ed. I. Gohberg), OT 33, Birkhäuser Verlag, Basel Boston Berlin, pp. 1-72, 1988.

[BGR2] J. A. Ball, I. Gohberg and L. Rodman, Two-sided Lagrange-Sylvester Interpolation Problems for Rational Matrix Functions, in *Proceeding Symposia in Pure Mathematics,* vol. 51, Part I, Amer. Math. Soc., Providence, pp. 17-83, 1990.

[BGR3] J. A. Ball, I. Gohberg and L. Rodman, *Interpolation of Rational Matrix Functions,* OT 45, Birkhäuser Verlag, Basel Boston Berlin, 1990.

[BR1] J. A. Ball and M. Rakowski, Zero-Pole Structure of Nonregular Rational Matrix Functions, in *Extension and Interpolation of Linear Operators and Matrix Functions*, OT 47, Birkhäuser Verlag, Basel Boston Berlin, pp. 137-193, 1990.

[BR2] J. A. Ball and M. Rakowski, Null-Pole Subspaces of Rectangular Rational Matrix Functions, *Linear Algebra and its Applications*, vol. 159, pp.81-120, 1991.

[BRW] J. Ball, M. Rakowski, B. Wyman, Coupling Operators, Wedderburn-Forney Spaces, and Generalized Inverses, to appear in *Linear Algebra and its Applications*, Special Issue, 1994.

[Fo] G. D. Forney, Jr., Minimal bases of rational vector spaces, with applications to multivariable linear systems, *SIAM Journal of Control*, vol. 13, no. 3, pp. 493-520, May 1975.

[CP] G. Conte and A. M. Perdon, Infinite Zero Module and Infinite Pole Module, *VII Int. Conf. Analysis and Opt. of Systems: Nice,* Lec. Notes Control and Inf. Science, Springer 62, pp. 302-315, 1984.

[GW] S. Giust and B. F. Wyman, The Global Observability Filtration for Singular Dynamical Systems, in *Proceedings of the International Symposium on Implicit and Nonlinear Systems*, Austin, Texas, Dec. 14-15, 1993, pp. 244-251.

[GLR] I. Gohberg, P. Lancaster and L. Rodman, *Matrix Polynomials*, Academic Press, New York London Paris, 1982.

[Ro] L. Rodman, Spectral Theory of Analytic Matrix Functions, PhD Dissertation, Tel-Aviv University, Tel-Aviv, Israel, June 1978.

[SWP] M. K. Sain, B. Wyman and J. Peczkowski, Extended Zeros and Model Matching, *SIAM J. Control Opt.*, vol. 29, pp. 562-593, 1991.

[WCP] B. Wyman, G. Conte and A. Perdon, Local and Global Linear System Theory, in *Frequency Domain and State Space Methods for Linear Systems*, C. Byrnes and A. Lindquist (editors), North Holland, Amsterdam, 1986.

[WS1] B. Wyman and M. K. Sain, The Zero Module and Essential Inverse Systems, *IEEE Trans. Circ. Systems*, vol. CAS-28, pp. 112-126, 1981.

[WS2] B. Wyman and M. K. Sain, On The Zeros of a Minimal Realization, *Lin. Alg. and its Appl.*, vol. 50 (1983) pp. 621-637.

[WSCP1] B. F. Wyman, M. K. Sain, G. Conte and A. M. Perdon, On the Zeros and Poles of a Transfer Function, *Lin. Alg. and its Appl.*, 122/123/124 (1989), 123-144.

[WSCP2] B. F. Wyman, M. K. Sain, G. Conte and A. M. Perdon, Poles and Zeros of Matrices of Rational Functions, *Linear Algebra and Its Applications*, vol. 157, pp. 113-139, 1991.

This research was partially supported by National Science Foundation Grant DMS-9302706.

A FRESH LOOK AT INVERSE DYNAMICAL SYSTEMS*

MICHAEL K. SAIN
Department of Electrical Engineering
University of Notre Dame
Notre Dame, Indiana 46556 USA

ABSTRACT

Thirteen years ago, polynomial module ideas were introduced into the systems literature to address the issue of zero structures in inverse dynamics (Wyman and Sain, 1981). Modules have since permitted a unified approach to the study of dynamical qualities for systems which are improper, nonminimal, implicit, and which display nontrivial kernels and cokernels. In this paper, we employ this tool to take a fresh look at the question of exactly how to make the zero structure of an inverse system as simple as possible. The issue turns out to be considerably more subtle than it would appear. We argue that the task of finding inverses with no more than the necessary zero structure does not have a natural solution for traditional cases of left and right inversion, unless the matrices involved are square. At first, this may seem quite counterintuitive. But, when we make clear just exactly how to account for all the zero structure of a system, then the true difficulty of the problem becomes strikingly clear.

1. INTRODUCTION

Many interesting questions arise in systems with unequal numbers of inputs and outputs, or in systems with a singular relationship between equal numbers of inputs and outputs. Among such questions, none is more intriguing than the disparity between total number of zeros and total number of poles. Indeed, when one makes a tally of the total number of poles and zeros of a transfer function matrix, with due attention both to the finite plane and the point at infinity, the number of zeros is not equal to the number of poles, as is so comfortably true for the single-input, single-output case.

It is natural to guess that nonzero kernels and cokernels have something to do with zero-like behavior. One fact in support of this view is that the total number of finite and infinite poles, regarded as points, is never less than the total number of finite and infinite zeros, regarded as points. But it is not an easy matter to envision a means to fold the kernel and cokernel information into an overall, integrated, pole-zero picture, which in the past has emphasized points, invariant factors, and so forth.

Nonetheless, there is a natural way to achieve the synthesis of the two types of information; and this is through the use of spaces to represent the poles and zeros. The reader will already be quite familiar with the usual state space idea and its associated operator A, which is traditionally associated with poles. Similar notions hold for zeros. From the notions of minimal polynomials, Cayley-Hamilton, and so forth, it is easy to surmise that polynomials in the operator A are quite natural scalars to employ in studying state spaces of poles and zeros. Thus, questions of poles and zeros may be quite naturally addressed in terms of spaces equipped with scalar-ring operations, namely modules. With regard to poles, such studies were initiated about thirty years ago (Kalman, 1965). For zeros, the corresponding steps are much more recent (Wyman and Sain, 1981).

In this paper, we employ this tool to take a fresh look at the question of exactly how to make the zero structure of an inverse system as simple as possible. As will become clear in the sequel, this issue is much more subtle than is generally appreciated. Moreover, it can fail to have a natural resolution in most surprising cases. We begin with a review of terminology and basic properties.

---

*This work was supported in part by the Frank M. Freimann Chair in Electrical Engineering at the University of Notre Dame.

2. BASIC POLE AND ZERO SPACES

For the field k, let $k[z]$ be the ring of polynomials in z with coefficients in k, and let $k(z)$ be the induced quotient field. Let R, U, and Y be finite-dimensional vector spaces over k. Observe that $k[z]$ and $k(z)$ are also k-vector spaces. Next use the k-bilinear tensor product to form $k[z]$-modules $R[z] = k[z] \otimes_k R$, $U[z] = k[z] \otimes_k U$, and $Y[z] = k[z] \otimes_k Y$, and $k(z)$-vector spaces $R(z) = k(z) \otimes_k R$, $U(z) = k(z) \otimes_k U$, and $Y(z) = k(z) \otimes_k Y$. These vector spaces serve as our spaces of signals in the discussion which follows. Note that this approach permits us to proceed without choosing bases. If $M(z) : R(z) \rightarrow U(z)$ is a $k(z)$-linear map, we wish to introduce in a precise way the spaces which will be used to capture the description of its poles and zeros.

The pole module $P(M(z))$ attached to a $k(z)$-linear map $M(z)$ is defined to be the $k[z]$-factor module $P(M(z)) = R[z]/\{R[z] \cap M^{-1}U[z]\}$, where M^{-1} is the inverse image function of $M(z)$, defined on $U[z]$ by $M^{-1}U[z] = \{r(z) : r(z) \in R(z) \text{ and } M(z)r(z) \in U[z]\}$. $P(M(z))$ is finitely generated as a $k[z]$-module, because $R[z]$ is finitely generated. Indeed, one can simply use a finite set of generators for the underlying vector space R for this purpose. It is a torsion module as well, because every element in $R[z]$, even though it is not itself in $M^{-1}U[z]$, can be scalar multiplied into one with an appropriately chosen polynomial $p(z)$ in $k[z]$.

Finitely generated, torsion modules over $k[z]$, such as the pole module $P(M(z))$, have the character of traditional state spaces of finite dimension over the field k. If we therefore consider it as a k-vector space, then $P(M(z)) = X_P$, which is the state space of a minimal realization of $M(z)$. The scalar multiplication in the module, given by the action $z : P(M(z)) \rightarrow P(M(z))$, defines a k-linear map $A_P : X_P \rightarrow X_P$ in the realization. This map, A_P, determines the pole dynamics of the transfer function matrix $M(z)$.

The definition of the pole module of $M(z)$ does not depend upon whether the kernel, ker $M(z)$, of $M(z)$, or the cokernel, coker $M(z)$, of $M(z)$ is nonzero. This is not in general the case for $k[z]$-zeros of $M(z)$. But there is an agreed-upon module containing those zeros of $M(z)$ which are finitely generated and torsion, namely those zeros which in their character are essentially like poles. These zeros tend to be the most familiar, and so we introduce them first. The finitely generated, torsion zero module $Z(M(z))$ attached to a $k(z)$-linear map $M(z)$ was introduced by Wyman and Sain (1981) as the $k[z]$-factor module $Z(M(z)) = \{M^{-1}U[z] + R[z]\}/\{\ker M(z) + R[z]\}$.

The zero module also permits a state-space interpretation. As a k-vector space, $Z(M(z)) = X_Z$; and it is not difficult to see that this space is of finite dimension. As in the case of the pole module, the scalar action $z : Z(M(z)) \rightarrow Z(M(z))$ defines a k-linear map $A_Z : X_Z \rightarrow X_Z$.

3. EXTENDED ZEROS

In our discussion of extending the notion of zeros of transfer function matrices, it will be convenient to consider two cases. The first case is associated with the matrix having a nontrivial kernel. Modules used to describe this case are of a character similar to that used by Kalman (1965) to discuss system outputs, and so we refer to these as zeros of output type. The second case is related to the matrix having a nontrivial cokernel. Modules used to describe the new zeros in this situation are of a character similar to that used in the Kalman theory for inputs, and so we refer to these as zeros of input type.

Consider first the zeros of output type. We construct the $k[z]$-factor module $Z_\Gamma(M(z)) = M^{-1}U[z]/\{R[z] \cap M^{-1}U[z]\}$, which is torsion. In the case in which ker $M(z)$ is not equal to zero, however, this module is not finitely generated. It will be called the Γ-zero module of $M(z)$. Immediately, we raise the question of the relation of this new module to that introduced in Section 2 for the traditional space of zeros. The answer to this question is quite intuitive and pleasing.

When ker $M(z)$ is not zero, we will define a new module $\Gamma(M(z))$ in the manner $\Gamma(M(z)) = \ker M(z)/\{R[z] \cap \ker M(z)\}$. The technique used to relate this new module to the traditional space of zeros makes use of a short exact sequence. It will be easier to explain this terminology when we have an explicit example at hand, and toward this end we introduce just such a sequence:

$$0 \to \Gamma(M(z)) \to Z_\Gamma(M(z)) \longrightarrow Z(M(z)) \to 0. \tag{3-1}$$

In (3-1), one needs to visualize above each arrow a linear mapping of modules. The picture is then one of a sequence of modules and linear mappings. Such a sequence is said to be exact at a particular module if the image of the incoming map is equal to the kernel of the outgoing map. The entire sequence is said to be exact if it is exact at each module in the sequence. Observe that exactness at the second module means that the second mapping has zero kernel; and in an analogous way, exactness at the fourth module in means that the third mapping has zero cokernel. Now we appeal to a technical point. The second module is both torsion and divisible, and so the center module is said to split, that is, it is isomorphic to a direct sum of the second and fourth modules, in the manner $Z_\Gamma(M(z)) \approx \Gamma(M(z)) \oplus Z(M(z))$.

This situation shows the strong intuitive character of the short exact sequence: If we wish to study the structure of a module, one way is to place it as the centerpiece in such a sequence. We will refer to $\Gamma(M(z))$ as the divisible zero module of $M(z)$, and to $Z_\Gamma(M(z))$, the Γ-zero module of $M(z)$, as an extended zero module. We can construct two intersecting short exact sequences, as shown in Figure 1. The horizontal sequence displays the two components of the extended zero module, while the vertical sequence explains further the nature of the submodule building block. There is an intuitive way of speaking about factor modules, such as that in the second last position in the vertical sequence. We can say that $\Gamma(M)$ consists of those vectors in the kernel of $M(z)$ which are not polynomial vectors.

$$\begin{tikzcd}
 & 0 \arrow[d] & & & \\
 & \ker M(z) \cap R[z] \arrow[d] & & & \\
 & \ker M(z) \arrow[d] & & & \\
0 \arrow[r] & \Gamma(M) \arrow[r] \arrow[d] & Z_\Gamma(M) \arrow[r] & Z(M) \arrow[r] & 0 \\
 & 0 & & &
\end{tikzcd}$$

$$\begin{tikzcd}
 & & & 0 \arrow[d] & \\
 & & & U[z] \cap \operatorname{im} M(z) \arrow[d] & \\
 & & & Y[z] \arrow[d] & \\
0 \arrow[r] & Z(M) \arrow[r] & Z_\Omega(M) \arrow[r] & \Omega(M) \arrow[r] \arrow[d] & 0 \\
 & & & 0 &
\end{tikzcd}$$

Figure 1. $Z_\Gamma(M) \approx \Gamma(M) \oplus Z(M)$ Figure 2. $Z_\Omega(G) \approx Z(G) \oplus \Omega(G)$

Now consider zeros of input type, by means of constructing the $k[z]$-factor module $Z_\Omega(M(z)) = U[z]/\{U[z] \cap MR[z]\}$. Although this module is finitely generated, it is not true that every equivalence class in it is torsion. We will refer to it as the Ω-zero module of $M(z)$. When coker $M(z)$ is nonzero, we can follow a program similar to that above, for the output zeros. If we define $\Omega(M(z)) = U[z]/\{U[z] \cap MR(z)\}$, then it may be established that there is a short exact sequence

$$0 \to Z(M(z)) \to Z_\Omega(M(z)) \to \Omega(M(z)) \to 0 \tag{3-2}$$

of $k[z]$-modules and $k[z]$-linear maps. The factor module in the sequence can be shown to be torsion-free. It then follows from the nature of the polynomial ring of scalars that $\Omega(M(z))$ is a free module, and from that it may be shown that the sequence splits, in the manner $Z_\Omega(M(z)) \approx \Omega(M(z)) \oplus Z(M(z))$. We will call $\Omega(M(z))$ the free zero module of $M(z)$. Like $Z_\Gamma(M(z))$, then, $Z_\Omega(M(x))$ is an extended zero module, this time of Ω-type. The same pictorial type of presentations can be made in this case as well, and we indicate one in Figure 2.

4. TRANSFER FUNCTION MATRIX EQUATIONS: FIXED ZEROS

Two theorems are reviewed in this section. Theorem 1 defines and characterizes the module of fixed zeros in the transfer function matrix equation $T(z) = P(z)M(z)$. Two cases are considered, depending upon whether $M(z)$ is given or whether $P(z)$ is given. The matrix $T(z)$ is assumed given in both cases. The modules of fixed zeros are given the name of matching zero modules. The two cases take on the character of the two foregoing types of extended zero modules. Theorem 2 shows in a precise technical sense that these matching zero modules are contained in all solutions to the transfer function matrix problem, in the one case as a factor module, and in the other case as a submodule.

The matching Γ-zero module is denoted by Z_Γ and defined to be the $k[z]$-factor module $Z_\Gamma = T^{-1}Y[z]/\{T^{-1}Y[z] \cap M^{-1}U[z]\}$, where $T(z) : R(z) \to Y(z)$ is a $k(z)$-linear map. The matching Ω-zero module is denoted by Z_Ω and defined to be the $k[z]$-factor module $Z_\Omega = \{PU[z] + TR[z]\}/TR[z]$, for $P(z) : U(z) \to Y(z)$ a $k(z)$-linear map. The algebraic character of matching zero modules of Γ-type and of Ω-type is settled by Theorem 1.

Theorem 1 (Matching Zeros, SWP-91)

Let Z_Γ and Z_Ω be the matching zero modules of Γ-type and Ω-type, respectively, for $k(z)$-linear maps $M(z) : R(z) \to U(z)$, $P(z) : U(z) \to Y(z)$, and $T(z) : R(z) \to Y(z)$. Then $Z_\Gamma \approx \Gamma \oplus Z'_\Gamma$, where Γ is torsion divisible and Z'_Γ is finitely generated and torsion; and $Z_\Omega \approx \Omega \oplus Z'_\Omega$, where Ω is finitely generated and free while Z'_Ω is finitely generated and torsion.

Remark

It can be seen quite clearly from the theorem that the two matching zero modules have the nature of the two types of extended zeros. We shall show in the next section that these modules have a most interesting structure. Theorem 2, which follows, then provides the inclusion results desired.

Theorem 2 (Fixed Zeros, SWP-91)

Suppose that Z_Γ and Z_Ω are the matching zero modules of Γ-type and Ω-type, respectively, and let $T(z) : R(z) \to Y(z)$ be a $k(z)$-linear map.

1. If $P(z) : U(z) \to Y(z)$ is a $k(z)$-linear map whose image contains that of $T(z)$, and if $M(z) : R(z) \to U(z)$ is a $k(z)$-linear map which satisfies the equation $T(z) = P(z)M(z)$, then there exists an epic, $k[z]$-linear map

$$\beta_\Omega(z) : Z_\Omega(M(z)) \to Z_\Omega, \tag{4-1}$$

so that Z_Ω is isomorphic to a factor module of the Ω-zero module of $M(z)$.

2. If $M(z) : R(z) \to U(z)$ is a $k(z)$-linear map whose kernel is contained in that of $T(z)$, and if $P(z) : U(z) \to Y(z)$ is a $k(z)$-linear map which satisfies the equation $T(z) = P(z)M(z)$, then there exists a monic, $k[z]$-linear map

$$\beta_\Gamma(z) : Z_\Gamma \to Z_\Gamma(P(z)), \tag{4-2}$$

so that Z_Γ is isomorphic to a submodule of the Γ-zero module of $P(z)$.

Remark

Equations (4-1) and (4-2) are the key assertions, the former an onto map, which makes the matching zero module a factor, and the latter a one-to-one map, which makes the matching zero module a submodule. These are just two basic ways of asserting inclusion in an algebraic sense.

5. INTUITIVE MEANING OF MATCHING ZEROS

In this section, we do two things. First, we present a technical result, Theorem 3, which breaks down the matching zero modules, in both cases, by means of triples of short exact sequences. Then, after the theorem, we provide a number of diagrams to help in its interpretation. These provide remarkable insights into the whole question of fixed zeros in particular, and more generally into the roles of kernels and cokernels in inverse dynamical system theory.

Theorem 3 (Matching Structure, SWP-91)

Let Z_Γ and Z_Ω be the matching zero modules of Γ-type and Ω-type, respectively, for $k(z)$-linear maps $M(z) : R(z) \to U(z)$, $P(z) : U(z) \to Y(z)$, and $T(z) : R(z) \to Y(z)$.

1. If $[T(z)\ P(z)] : R(z) \oplus U(z) \to Y(z)$ is the $k(z)$-linear map with action
$$[T(z)\ P(z)](r(z), u(z)) = T(z)r(z) + P(z)u(z), \tag{5-1}$$
then there exist $k[z]$-modules Z_1 and P_1, together with appropriate $k[z]$-linear maps, such that the following three short sequences are exact:
$$0 \to P(T(z)) \to P([T(z)\ P(z)]) \to P_1 \to 0; \tag{5-2a}$$
$$0 \to Z_1 \to Z_\Omega(T(z)) \to Z_\Omega([T(z)\ P(z)]) \to 0; \tag{5-2b}$$
$$0 \to Z_1 \to Z_\Omega \to P_1 \to 0. \tag{5-2c}$$

2. If
$$\begin{bmatrix} T(z) \\ M(z) \end{bmatrix} : R(z) \to Y(z) \oplus U(z) \tag{5-3a}$$
is the $k(z)$-linear map having action
$$\begin{bmatrix} T(z) \\ M(z) \end{bmatrix} : (r(z)) = (T(z)r(z), M(z)r(z)), \tag{5-3b}$$
then there exist $k[z]$-modules Z_2 and P_2, together with appropriate $k[z]$-linear maps, such that the following three short sequences are exact:
$$0 \to P_2 \to P\left(\begin{bmatrix} T(z) \\ M(z) \end{bmatrix}\right) \to P(T(z)) \to 0; \tag{5-4a}$$
$$0 \to Z_\Gamma\left(\begin{bmatrix} T(z) \\ M(z) \end{bmatrix}\right) \to Z_\Gamma(T(z)) \to Z_2 \to 0; \tag{5-4b}$$
$$0 \to P_2 \to Z_\Gamma \to Z_2 \to 0. \tag{5-4c}$$

A diagrammatic interpretation for the second part of the representation theorem is given in the next three figures. The first part of the theorem can also receive such a treatment. For the sake of space, however, we leave that for an exercise to the reader. Figure 3 shows the basic short exact sequence for the fixed module Z_Γ, together with the further breakdown of the submodule P_2.

$$\begin{tikzcd}
 & 0 \arrow[d] & & & \\
0 \arrow[r] & P_2 \arrow[r] \arrow[d] & Z_\Gamma \arrow[r] & Z_2 \arrow[r] & 0 \\
 & P\left(\begin{bmatrix} T \\ M \end{bmatrix}\right) \arrow[d] & & & \\
 & P(T) \arrow[d] & & & \\
 & 0 & & &
\end{tikzcd}$$

Figure 3.

$$\begin{tikzcd}
 & & & 0 \arrow[d] & \\
 & & & Z_\Gamma\left(\begin{bmatrix} T \\ M \end{bmatrix}\right) \arrow[d] & \\
 & & & Z_\Gamma(T) \arrow[d] & \\
0 \arrow[r] & P_2 \arrow[r] & Z_\Gamma \arrow[r] & Z_2 \arrow[r] \arrow[d] & 0 \\
 & & & 0 &
\end{tikzcd}$$

Figure 4.

The striking thing about Figure 3 is the appearance of the composite matrices built from $T(z)$ and $M(z)$. This behavior, as well as the appearance of the extended zero spaces, is quite unpredicted from the single-input, single-output case. Figure 4 then supplies the corresponding information for the factor module Z_2. An intuitive interpretation of these two diagrams, appears in Figure 5.

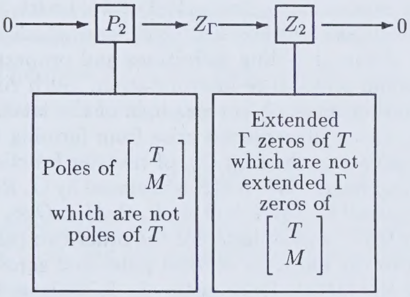

0
P_2
Z_Γ
Z_2
0
Poles of $\begin{bmatrix} T \\ M \end{bmatrix}$ which are not poles of T
Extended Γ zeros of T which are not extended Γ zeros of $\begin{bmatrix} T \\ M \end{bmatrix}$

Figure 5.

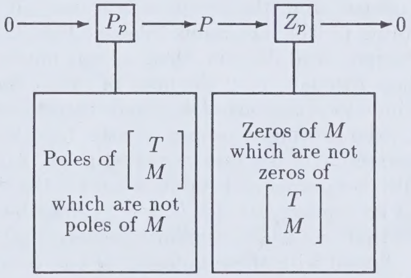

0
P_p
P
Z_p
0
Poles of $\begin{bmatrix} T \\ M \end{bmatrix}$ which are not poles of M
Zeros of M which are not zeros of $\begin{bmatrix} T \\ M \end{bmatrix}$

Figure 6.

In the next section, we report briefly, so that the reader can compare, the original results of Conte, Perdon, and Wyman (1988) on fixed poles.

6. CONTE, *ET. AL.* (1988): FIXED POLES

We omit the technical theorem, and make use of the diagrammatic method to indicate the qualitative nature of the results. For purposes of consistency, the fixed poles are considered for the same case as that of Figures 3, 4, and 5. There is a module of fixed poles, P, which may be placed into the center of a short exact sequence as in Figure 7, wherein the nature of the submodule is also examined.

$$\begin{tikzcd}
& 0 \arrow[d] & & & \\
0 \arrow[r] & P_p \arrow[r] \arrow[d] & P \arrow[r] & Z_p \arrow[r] & 0 \\
& P\left(\begin{bmatrix} T \\ M \end{bmatrix}\right) \arrow[d] & & & \\
& P(M) \arrow[d] & & & \\
& 0 & & &
\end{tikzcd}$$

Figure 7.

$$\begin{tikzcd}
& & & 0 \arrow[d] & \\
& & & Z\left(\begin{bmatrix} T \\ M \end{bmatrix}\right) \arrow[d] & \\
& & & Z(M) \arrow[d] & \\
0 \arrow[r] & P_p \arrow[r] & P \arrow[r] & Z_p \arrow[r] \arrow[d] & 0 \\
& & & 0 &
\end{tikzcd}$$

Figure 8.

Figure 8 then further explores the character of the factor module in the horizontal sequence. Next compare Figures 3 and 7. At first, the vertical sequences may seem to be the same. But note that the pole modules at the bottom of the vertical sequences are different, one depending upon $T(z)$ and the other depending upon $M(z)$. Figure 8 is to be compared with Figure 4. Again, there is the difference between $T(z)$ and $M(z)$; but note also the extension of the notion of zero. Finally, there is Figure 6, which is once more the intuitive counterpart of a joining together of Figures 7 and 8. We suspect that the reader can guess the result for fixed poles in the other case.

7. THE POINT AT INFINITY

Up to this point, we have put the focus on the polynomial ring $k[z]$, so that as a result the zeros and poles of traditional type have been in the finite plane, so to speak—because we have not been too specific about the nature of k. The terminology comes, of course, from the complex field case. If we replace the subring $k[z]$ of $k(z)$ with any subring O which contains the base field k, which has quotient field $k(z)$, and which is a principal ideal domain, then a great number of the preceding definitions and properties follow through, with the need of course for some physical re-interpretation. Such rings include localizations of $k[z]$ and discrete valuation rings. A key example of the latter is O_∞, the subring of proper transfer functions. Other illustrations arise from forming the intersection of discrete valuation rings. An instance is the ring O_{ps} of transfer functions which are proper and stable, when k is the real numbers. When $k[z]$ is replaced by O, $R[z]$, and its counterparts for U and Y, must be replaced by $\Omega_O R = O \otimes_k R$, $\Omega_O U = O \otimes_k U$,, and $\Omega_O Y = O \otimes_k Y$. Similarly, replace $U[z]$ by $\Omega_O U$, and so forth for the other two cases.

Armed with these notions, we can return to the question of fixed poles and zeros in the transfer function matrix equation $T(z) = P(z)M(z)$. Once again the focus is on the situation in which $M(z)$ is given. The other case is left as an exercise. The appropriate pictures, then, for the fixed poles and the fixed zeros are those of Figures 9 and 10. Notice that the superscript ∞ has been added to distinguish these results. Recall that, if there are no fixed poles at infinity, then it will be possible to obtain a causal or proper solution.

8. THE EXTENDED PLANE

We have observed in the preceding discussion that the number of finite poles and poles at infinity of a transfer function $G(z)$ may be greater than the number of its finite zeros and zeros at infinity. This difference between the total number of poles and the total number of zeros may be calculated in terms of Kronecker indices or Wedderburn numbers (Wedderburn, 1934; Forney, 1975).

$$\begin{tikzcd}
 & & & 0 \arrow[d] & \\
 & & & Z^{\infty}\left(\begin{bmatrix} T \\ M \end{bmatrix}\right) \arrow[d] & \\
 & 0 \arrow[d] & & Z^{\infty}(M) \arrow[d] & \\
0 \arrow[r] & P_p^{\infty} \arrow[r] \arrow[d] & P^{\infty} \arrow[r] & Z_p^{\infty} \arrow[r] \arrow[d] & 0 \\
 & P^{\infty}\left(\begin{bmatrix} T \\ M \end{bmatrix}\right) \arrow[d] & & 0 & \\
 & P^{\infty}(M) \arrow[d] & & & \\
 & 0 & & &
\end{tikzcd}$$

Figure 9.

$$\begin{tikzcd}
 & & & 0 \arrow[d] & \\
 & & & Z_{\Gamma}^{\infty}\left(\begin{bmatrix} T \\ M \end{bmatrix}\right) \arrow[d] & \\
 & 0 \arrow[d] & & Z_{\Gamma}^{\infty}(T) \arrow[d] & \\
0 \arrow[r] & P_2^{\infty} \arrow[r] \arrow[d] & Z^{\infty} \arrow[r] & Z_2^{\infty} \arrow[r] \arrow[d] & 0 \\
 & P^{\infty}\left(\begin{bmatrix} T \\ M \end{bmatrix}\right) \arrow[d] & & 0 & \\
 & P^{\infty}(T) \arrow[d] & & & \\
 & 0 & & &
\end{tikzcd}$$

Figure 10.

The Wedderburn-Forney construction introduced in (Wyman *et al.* 1989) attaches a finite dimensional vector space over k, $\mathcal{W}(\mathcal{C})$, to a $k(z)$-vector space $\mathcal{C}(z)$. This space associated with $\mathcal{C}(z) \subset V(z)$ is $\mathcal{W}(\mathcal{C}) = \pi_-(\mathcal{C})/\{\mathcal{C} \cap z^{-1}\Omega_\infty V\}$, where π_- is the $k-$linear projection which discards the polynomial part of a vector.

It is possible to combine the traditional pole space, associated with the finite plane, with its counterpart at infinity, by forming a direct sum. We call this the global pole space and denote it by $\mathcal{P}(G(z))$. Likewise, we can form a global zero space from the direct sum of the traditional zero space and its counterpart at infinity, and we denote the result by $\mathcal{Z}(G(z))$. The following theorem then holds.

Theorem 4 (Global Structure, Wyman, *et al.*, 1989)

There is an exact sequence of finite dimensional vector spaces over k,

$$0 \to \mathcal{Z}(G(z)) \to \frac{\mathcal{P}(G(z))}{\mathcal{W}(ker\ G(z))} \to \mathcal{W}(im\ G(z)) \to 0, \tag{8-1}$$

where $\mathcal{Z}(G(z))$ is the global space of zeros of $G(z)$, $\mathcal{P}(G(z))$ is the global space of poles of $G(z)$, and $\mathcal{W}(\ )$ is the Wedderburn-Forney construction.

Remark

Using the traditional dimension relations for quotient spaces, we can see easily that the two Wedderburn-Forney spaces in this sequence account for the missing zeros, which then balance with the total number of poles, as desired. We have, then, another way of looking at kernels and cokernels in the pole-zero setting.

9. INVERTIBILITY: A FRESH LOOK

In this section, our main results involve a further and more detailed examination of the transfer function equation $T(z) = P(z)M(z)$ under two conditions: (1) ker $T(z) = 0$ and (2) coker $T(z) = 0$. The big question is whether or not the fixed-zero constraints represented by (4-1) and (4-2), in Theorem 2, can be achieved with isomorphism. In other words, are there solutions $M(z)$, or $P(z)$, which have the matching zero modules

Z_Ω, or Z_Γ, as their extended zero modules of type Ω, or Γ, respectively? We refer to such solutions as *essential solutions.* Because of space limitations, it will be necessary to forego the proofs.

Because essential solutions must produce the correct free or divisible extended zeros, the next theorem provides a basis to settle the question completely.

Theorem 5 (Zeros: Free, Divisible, and Fixed)

In the equation $T(z) = P(z)M(z)$ of $k(z)$-linear maps, $\Omega(M(z)) \approx \Omega(Z_\Omega)$, and $\Gamma(P(z)) \approx \Gamma(Z_\Gamma)$, if and only if $M(\ker T(z)) = \ker P(z)$.

Remark

If a system $T(z)$ is invertible, then obviously $\ker T(z) = 0$. Thus, to achieve the constraints of Theorem 2 with isomorphism implies that $\ker P(z) = 0$ as well, which in turn would require that $P(z)$ is an isomorphism. Then $M(z)$ is an isomorphism as well, so that there are no extended zeros of either type anywhere in the equation. Therefore, for the more general case of left inversion or right inversion, when the factors are not isomorphisms, there is no possibility of constructing solutions whose extended zero modules meet the bounds established in Theorem 2. We will thus be able to have a conclusive theorem. First, however, there are more remarks.

Remark

It may seem that the linking of the conditions $\Omega(M(z)) \approx \Omega(Z_\Omega)$ and $\Gamma(P(z)) \approx \Gamma(Z_\Gamma)$ in the theorem could be restrictive. In other words, we could envision a situation in which the first condition would hold, but not the second, or conversely. But it may be shown that this is not the case. Indeed, if the first condition holds, then so does the second; and if the second condition holds, then so does the first. Accordingly, no loss of generality is incurred.

Remark

Further, the reader should observe that Theorem 5 is not speaking of essential solutions, but only of a necessary condition for essential solutions. It is possible for the condition $M(\ker T(z)) = \ker P(z)$ to occur without both $P(z)$ and $M(z)$ being essential solutions. Now the theorem.

Theorem 6 (Invertibility and Fixed Zeros)

Consider the general invertibility equations $1_{Y(z)} = P(z)M(z)$ or $1_{R(z)} = P(z)M(z)$. Then the factors $M(z)$ or $P(z)$ are essential solutions if and only if they are $k(z)$-linear isomorphisms, in which case both factors share this property.

In view of the very stringent nature of Theorem 5, we may ask about the situation under slightly weaker conditions. So we shall consider $k(z)$-linear maps $T(z)$ which have zero kernels or cokernels, but not necessarily both. For the former case, we refer to *extended left invertibility.* For the latter case, we speak of *extended right invertibility.* The next theorem addresses our question in these circumstances.

Theorem 7 (Extended Invertibility and Fixed Zeros)

Consider the equation $T(z) = P(z)M(z)$ of $k(z)$-linear maps. If a system $P(z)$ is right invertible, and if $\operatorname{coker} T(z) = 0$, then there exists a solution $M(z)$ which is essential if and only if the dimension of R is no less than the dimension of U. If a system $M(z)$ is left invertible, and if $\ker T(z) = 0$, then there exists a solution $P(z)$ which is essential if and only if the dimension of Y is no less than the dimension of U.

Remark

Notice that, in this theorem, if we add to the first part the assumption that $\ker T(z)$ vanishes, then clearly the discussion reduces to that of Theorem 6. Thus Theorem 7 is a true generalization.

Remark

A number of further generalizations are possible; but space limitations do not permit their elaboration. For instance, we can generalize Theorem 7 to the situations in which im $T(z)$ =im $P(z)$ on the one hand, and ker $T(z)$ = ker $M(z)$ on the other.

10. CONCLUSIONS

We have seen in Section 8 that an accounting of all the zeros in a multi-input, multi-output system—so as to have the total number of zeros equal to the total number of poles—leads naturally to the study of kernels and cokernels of the transfer functions involved. However, Theorem 4 is valid on the k-vector space level only, because the representation of poles and zeros at infinity, and their representation elsewhere, require different rings. If one wishes to involve dynamical action in the discussion, and thus employ module theory, it is necessary to develop the kernel, or cokernel, into a module. This means, in intuitive terms, that we must find a larger set which includes the kernel, or cokernel, and which is closed under module scalar multiplication. It is in just this way that the notions of divisible and free zeros have been brought into the picture. When combined with the classical transmission zero modules, they lead to the extended zero modules of Section 3. Then the study of fixed zeros in transfer function equations is possible and natural, as presented in Sections 4 and 5. Moreover, many questions which pertain to invertibility can be understood in this format. Remarkably, it turns out that the problem of constructing left inverses or right inverses with the simplest zero structure possible cannot be solved unless the systems are isomorphisms. This means that the design of zeros for inverse systems will be more of an *ad hoc* procedure, depending upon what structures are chosen to be combined with the fixed zeros. In view of the importance of inversion in communication and control, this is quite a thought-provoking situation.

REFERENCES

G. Conte, A.M. Perdon, and B.F. Wyman, "Fixed Poles in Transfer Function Equations", *SIAM Journal on Control and Optimization*, Volume 26, pp. 356-368, 1988.

G.D. Forney, Jr., "Minimal Bases of Rational Vector Spaces with Applications to Multivariable Linear Systems", *SIAM Journal on Control and Optimization*, Volume 13, pp. 493-520, 1975.

R.E. Kalman, "Algebraic Structure of Linear Dynamical Systems. I. The Module of Σ", *Proceedings National Academy of Sciences (USA)*, Volume 54, pp. 1503-1508, 1965.

M.K. Sain, B.F. Wyman, and J.L. Peczkowski, "Extended Zeros and Model Matching", *SIAM Journal on Control and Optimization*, Volume 29, pp. 562-593, 1991.

J.H.M. Wedderburn, *Lectures on Matrices.* American Mathematical Society Colloquium Publications, Volume 17, Chapter 4, 1934.

B.F. Wyman and M.K. Sain, "The Zero Module and Essential Inverse Systems", *IEEE Transactions on Circuits and Systems*, Volume 27, pp. 112-126, 1981.

B.F. Wyman, M.K. Sain, G. Conte, and A.M. Perdon, "On the Zeros and Poles of a Transfer Function", *Journal of Linear Algebra and Its Applications*, Volumes 122/123/124, pp. 123-144, 1989.

The Global Controllability Filtration for Improper Transfer Functions and the Wedderburn-Forney Construction

Steven J. Giust
Department of Mathematics
The Ohio State University
Columbus, OH 43210
email: sgiust@magnus.acs.ohio-state.edu

Abstract Controllability and observability indices for a singular system coincide with the minimal indices associated with certain Wedderburn-Forney spaces. We define filtrations on these Wedderburn-Forney Spaces that are a natural extension of the classical controllability and observability filtrations of a strictly proper transfer function. The controllability results agree with work done by Michel Malabre, and via duality we establish how to extend these methods to the observability case.

1 Notation

Let k be a field and $k(z)$ be the corresponding field of rational functions. We denote by $\mathcal{O}_\infty$ the local ring at infinity of proper rational functions. Let U, X, and Y be, respectively, the usual input, state, and output vector spaces associated with an $(\hat{E}, \hat{A}, \hat{B}, \hat{C})$ state space realization

$$\begin{aligned} \hat{E}x(i+1) &= \hat{A}x(i) + \hat{B}u(i) \\ y &= \hat{C}x(i) \end{aligned} \tag{1}$$

of an arbitrary $p \times m$ transfer function $G(z)$ with entries from $k(z)$. Let V be any finite dimensional k-vector space. We will denote by $V(z)$ the $k(z)$-module $V \otimes_k k(z)$, by $V[z]$ or ΩV the $k[z]$-module $V \otimes_k k[z]$, by $\Omega_\infty V$ the $\mathcal{O}_\infty$ -module $V \otimes_k \mathcal{O}_\infty$, and by $z^{-1}\Omega_\infty V$ or $V(z)_{sp}$ the $\mathcal{O}_\infty$ -module $V \otimes_k z^{-1}\mathcal{O}_\infty$. Viewed more concretely, $V(z)$ consists of column vectors of rational functions, ΩV consists of polynomial column vectors, $\Omega_\infty V$ consists of vectors of proper rational functions, and $z^{-1}\Omega_\infty V$ consists of vectors of strictly proper rational functions. Each module is free over its respective PID and finitely generated. We will write a function $f \in k(z)$ in terms of its polynomial (poly) and strictly proper (sp) parts as $f = \pi_+(f) + \pi_-(f)$, where $\pi_+(f) \in k[z]$ and $\pi_-(f) \in z^{-1}\mathcal{O}_\infty$. Given a vector $\vec{v}(z)$ in any of the above mentioned modules, we extend this map componentwise so that $\vec{v}(z) = \pi_+(\vec{v}(z)) + \pi_-(\vec{v}(z))$, where $\pi_+(\vec{v}(z)) \in \Omega V$ and $\pi_-(\vec{v}(z)) \in z^{-1}\Omega_\infty V$.

Our primary interest is when $\hat{E}$ is singular. As in [1], we will always assume $(\hat{E}z - \hat{A})^{-1}$ exists. We view $G(z) = G_{poly} + G_{sp}$ with $G_{sp} = C_0(zI - A)^{-1}B_0$, $G_{poly} = C_\infty(zJ - I)^{-1}B_\infty$, and so

$$\begin{aligned} G(z) &= \left[\, C_\infty \mid C_0 \,\right] \left[z \left(\begin{array}{c|c} J & 0 \\ \hline 0 & I \end{array} \right) - \left(\begin{array}{c|c} I & 0 \\ \hline 0 & A \end{array} \right) \right]^{-1} \left[\begin{array}{c} B_\infty \\ \hline B_0 \end{array} \right] && (2) \\ &= \hat{C} \left[z\hat{E} - \hat{A} \right]^{-1} \hat{B}. && (3) \end{aligned}$$

Given $G(z) : U(z) \to Y(z)$ we attach various pole and zero spaces to $G(z)$. The finite pole and transmission zero $k[z]$-modules are defined, respectively, as

$$X(G) = \frac{\Omega U}{G^{-1}(\Omega Y) \ \cap \ \Omega U} \tag{4}$$

$$Z(G) = \frac{G^{-1}(\Omega Y) \ + \ \Omega U}{\ker G(z) \ + \ \Omega U}. \tag{5}$$

$X(G)$ consists of polynomial inputs modulo polynomial inputs that produce polynomial outputs. The polynomial elements in $X(G)$ are represented as $\vec{\mathbf{u}}[z]$. This module was the first constructed and follows the Kalman input output construction in [9]. Their analogs at infinity are the pole and zero $\mathcal{O}_\infty$ -modules defined as

$$X_\infty(G) = \frac{z^{-1}\Omega_\infty U}{G^{-1}(z^{-1}\Omega_\infty Y) \ \cap \ z^{-1}\Omega_\infty U} \tag{6}$$

$$Z_\infty(G) = \frac{G^{-1}(z^{-1}\Omega_\infty Y) \ + \ z^{-1}\Omega_\infty U}{\ker G(z) \ + \ z^{-1}\Omega_\infty U}. \tag{7}$$

$X_\infty(G)$ consists of strictly proper inputs modulo strictly proper inputs that produce strictly proper outputs. The strictly proper elements in $X_\infty(G)$ are represented as $\vec{\mathbf{v}}_{sp}(z)$. Finally, the realization pole space $\mathcal{O}_\infty$ -module is defined as

$$X'_\infty(G) = \frac{\Omega_\infty U}{G^{-1}(z^{-1}\Omega_\infty Y) \ \cap \ \Omega_\infty U}. \tag{8}$$

These modules have been studied in earlier works such as [2, 3, 4, 10, 11, 12]. Each module is finite dimensional over k. These spaces may be assembled into global spaces by direct summation as $\mathcal{X}(G) = X(G) \oplus X_\infty(G)$, $\mathcal{X}'(G) = X(G) \oplus X'_\infty(G)$, and $\mathcal{Z}(G) = Z(G) \oplus Z_\infty(G)$. The global spaces carry no natural module structure since they are direct sums of modules over different rings [1, 2, 3, 4].

Classically, $X(G) \overset{v.sp.}{\cong} \langle [B_o | AB_o | \ldots | A^{n-1}B_o] \rangle_k$ and is the realization space for G_{sp}. Similarly, $X'_\infty(G)$ is the classical realization space for G_{poly}. There exist vectors $\{\vec{\mathbf{f}}_\mathbf{i}\}_{i=1}^m \subseteq \Omega_\infty U$ and integers $e_1 \geq e_2 \geq \ldots \geq e_m$ such that

$$X'_\infty(G) \overset{module}{\cong} \frac{\vec{\mathbf{f}}_1}{(z^{-1})^{e_1+1}} \oplus \ldots \oplus \frac{\vec{\mathbf{f}}_m}{(z^{-1})^{e_m+1}} \overset{v.sp.}{\cong} [B_\infty | JB_\infty | \ldots | J^{e_1}B_\infty] \tag{9}$$

where multiplication by z^{-1} in the module structure is equivalent to multiplication by the nilpotent matrix J in the vector space structure [1]. From the definition of $X_\infty(G)$ we see that

$$X_\infty(G) \cong \frac{z^{-1}\vec{\mathbf{f}}_1}{(z^{-1})^{e_1+1}} \oplus \ldots \oplus \frac{z^{-1}\vec{\mathbf{f}}_m}{(z^{-1})^{e_m+1}} \cong JX'_\infty(G). \tag{10}$$

That is to say

$$X_\infty(G) \oplus X(G) \ = \ \hat{E}(X'_\infty(G) \oplus X(G)) \tag{11}$$

$$\text{and so } \mathcal{X}(G) \ \subseteq \ \operatorname{im}\hat{E}. \tag{12}$$

We will need this result later.

2 The Fundamental Pole-Zero Exact Sequence and the Wedderburn-Forney Construction

A full accounting of the zeros of a transfer function must take into consideration not only the global zero space $\mathcal{Z}(G)$ but also the "generic zeros," which arise from the kernel and cokernel of G. Following [3, 4] we measure these generic zeros by the dimension over k of finite dimensional *Wedderburn-Forney Vector Spaces* (WFS) constructed from the kernel and cokernel of G.

Given $V(z)$, finite dimensional over $k(z)$, and $\mathcal{C} \subseteq V(z)$, the WFS of $\mathcal{C}$ may be defined in several ways two of which are

$$W(\mathcal{C}) = \frac{\pi_+(\mathcal{C})}{\pi_+(\mathcal{C}) \cap (\mathcal{C})} \text{ and} \tag{13}$$

$$W(\mathcal{C}) = \frac{\pi_-(\mathcal{C})}{\pi_-(\mathcal{C}) \cap (\mathcal{C})}. \tag{14}$$

For example, let

$$\vec{\mathbf{c}} = \begin{pmatrix} -z+2+3z^{-1} \\ 3z^2+z^{-1}+4z^{-2} \end{pmatrix} \in \mathcal{C}$$

$$\text{then } \pi_+(\vec{\mathbf{c}}) = \begin{pmatrix} -z+2 \\ 3z^2 \end{pmatrix} \text{ so } \overline{\begin{pmatrix} -z+2 \\ 3z^2 \end{pmatrix}} \in W(\mathcal{C}).$$

That is, definition (13) consists of polynomial parts of vectors from $\mathcal{C}$ modulo those vectors in $\mathcal{C}$ that are themselves polynomial. The main result of [4] relates the global poles and all zeros of the transfer function through the following *Fundamental Pole-Zero Exact Sequence* (FPZES)

$$0 \to \mathcal{Z}(G) \xrightarrow{\alpha} \frac{X(G) \oplus X_\infty(G)}{W(\ker[G])} \xrightarrow{\beta} W(\text{im } [G]) \to 0. \tag{15}$$

The sequence and spaces play a crucial role in our treatment of global controllability and observability indices. In this work our interest rests primarily with the latter half of the sequence and with the map β where if $(\vec{\mathbf{u}}[z],\ \vec{\mathbf{v}}_{sp}(z)) \in \mathcal{X}(G)$, then

$$\beta(\vec{\mathbf{u}}[z],\ \vec{\mathbf{v}}_{sp}(z)) \overset{def}{=} \pi_+ \overline{\left(\begin{bmatrix} I \\ G \end{bmatrix} (\vec{\mathbf{u}}[z] + \vec{\mathbf{v}}_{sp}(z)) \right)} \in W(\text{im} \begin{bmatrix} I \\ G \end{bmatrix}). \tag{16}$$

Lemma 1 *Given G and a Right Globally Coprime Matrix Fraction Decomposition (RGCMFD) pair of polynomial matrices (N, D) such that $G = ND^{-1}$, let $\vec{d}_i$ be the i^{th} column of $\begin{bmatrix} D \\ N \end{bmatrix}$ and e_i its corresponding column degree. Then we have*

(i) $X(G) = X(\begin{bmatrix} I \\ G \end{bmatrix})$ and $X_\infty(G) = X_\infty(\begin{bmatrix} I \\ G \end{bmatrix})$.

(ii) $\ker\begin{bmatrix} I \\ G \end{bmatrix} = 0 \Rightarrow W(\ \ker\begin{bmatrix} I \\ G \end{bmatrix}) = 0.$

(iii) $\mathcal{Z}(\begin{bmatrix} I \\ G \end{bmatrix}) \ = \ 0.$

(iv) $W(\mathrm{im}\begin{bmatrix} I \\ G \end{bmatrix}) = W(\mathrm{im}\begin{bmatrix} D \\ N \end{bmatrix})$ *with* $\dim_k(\ W(\mathrm{im}\begin{bmatrix} D \\ N \end{bmatrix})\ ) = e \overset{def}{=} \sum_{i=1}^{m} e_i.$

(v) *((i), (ii), (iii), and (iv))* $\Rightarrow X(G) \oplus X_\infty(G) \overset{\beta}{\cong} W(\mathrm{im}\begin{bmatrix} I \\ G \end{bmatrix}) = W(\mathrm{im}\begin{bmatrix} D \\ N \end{bmatrix}).$

(vi) $W(\mathrm{im}\begin{bmatrix} D \\ N \end{bmatrix}) = \langle \mathcal{B} \rangle_k$ *where* $\mathcal{B} = \{\ \overline{\pi_+(z^{-j}\vec{\mathbf{d}}_i)}\ | i = 1, \ldots, m;\ j = 1, \ldots, e_i\}$. *That is* $\mathcal{B}$ *forms a basis over* k *for* $W(\mathrm{im}\begin{bmatrix} D \\ N \end{bmatrix})$.

Proof: (i) Let $U[z] \in \Omega U$. $U[z] \in (G^{-1}(\Omega Y) \cap \Omega U) \Leftrightarrow [(U[z] \in \Omega U) \wedge (G(U[z]) \in \Omega Y)] \Leftrightarrow \left(\begin{bmatrix} I \\ G \end{bmatrix} U[z] \in \begin{pmatrix} \Omega U \\ \Omega Y \end{pmatrix}\right) \wedge (U[z] \in \Omega U)$. Similarly, $X_\infty(G) \ = \ X_\infty(\begin{bmatrix} I \\ G \end{bmatrix})$. So $X(G) \ \oplus\ X_\infty(G) \ = X(\begin{bmatrix} I \\ G \end{bmatrix}) \ \oplus\ X_\infty(\begin{bmatrix} I \\ G \end{bmatrix})$.

(ii) Clear.

(iii) We show $Z(\begin{bmatrix} I \\ G \end{bmatrix}) = 0$. The proof $Z_\infty(\begin{bmatrix} I \\ G \end{bmatrix}) = 0$ is similar. Let $\vec{\mathbf{u}}(z) \in Z(\begin{bmatrix} I \\ G \end{bmatrix})$. Then $\begin{bmatrix} I \\ G \end{bmatrix} \vec{\mathbf{u}}(z) = \vec{\mathbf{y}}[z] \in \Omega Y \Rightarrow \vec{\mathbf{u}}(z) \in \Omega U$ because of the identity matrix component. Then $\vec{\mathbf{u}}(z) \equiv \vec{\mathbf{0}}$. in $Z(\begin{bmatrix} I \\ G \end{bmatrix})$.

(iv) Note $\begin{bmatrix} I \\ G \end{bmatrix} = \begin{bmatrix} D \\ N \end{bmatrix} D^{-1}$ so that $\mathrm{im}\begin{bmatrix} I \\ G \end{bmatrix} = \mathrm{im}\begin{bmatrix} D \\ N \end{bmatrix}$. The Wedderburn construction is a function of the image space and is independent of choice of basis.

(v) Apply (i), (ii), (iii), and (iv) to equation (15). The result follows immediately.

(vi) See [4, 8]; proof follows later. □

3 Filtrations

Define $s = (\max_i(\{e_i\}) - 1) \geq 0$. The π_+-Wedderburn filtration (π_+WF) is defined as $W(\mathrm{im}\begin{bmatrix} D \\ N \end{bmatrix}) = W_s \supseteq \ldots \supseteq W_t \supseteq \ldots W_0 \supseteq (0) = W_{-1}$ where

$$W_t = \overline{\pi_+(\mathrm{im}\begin{bmatrix} D \\ N \end{bmatrix}) \cap (\vec{\mathbf{k}}_0 + z\vec{\mathbf{k}}_1 + \ldots + z^t\vec{\mathbf{k}}_t)}. \tag{17}$$

That is, $[w] \in W_i$ iff $[w]$ has a representative of degree i or less in $W(\mathrm{im}\begin{bmatrix} D \\ N \end{bmatrix})$.

Now define

$$\mathcal{B}_t = \{\pi_+(z^{-j}\vec{\mathbf{d}}_i) | 0 \leq (e_i - j) \leq t; j, e_i > 0; 1 \leq i \leq m\}. \tag{18}$$

We have $W_t \cong \langle \mathcal{B}_t \rangle_k$. To prove this we will first need another lemma.

Lemma 2 (Predictable Degree) *Let* $\begin{bmatrix} D \\ N \end{bmatrix}$ *and the* $\dim(W(\mathrm{im}\begin{bmatrix} D \\ N \end{bmatrix})) = e$ *be as in Lemma 1. Let* $\vec{\mathbf{y}}[z] \neq 0$ *be an arbitrary vector in* $\langle \mathcal{B} \rangle_k$, *say* $\vec{\mathbf{y}}[z] = \sum_{s=1}^{e} a_s \vec{\mathbf{b}}_s$, $a_s \in k$. *Then* $\deg(\vec{\mathbf{y}}[z]) = \max_{a_s \neq 0}(\deg \vec{\mathbf{b}}_s), 1 \leq s \leq e$.

Proof: Suppose first that $\vec{\mathbf{h}}(z) = \sum_{i=1}^{m} a_i(z)\vec{\mathbf{d}}_i[z] = \begin{bmatrix} D \\ N \end{bmatrix}(\vec{\mathbf{a}})$, $a_i(z) = \frac{n_i[z]}{d_i[z]} \in k(z)$. Define $d[z] = \mathrm{lcm}\{d_i[z]\}$ and $\deg(d[z]) = l$. Then $d[z]\vec{\mathbf{h}}(z) = \vec{\mathbf{h}}'[z] = \begin{bmatrix} D \\ N \end{bmatrix}\vec{\mathbf{a}}'[z]$ is polynomial. In [6] we see $\deg(\vec{\mathbf{h}}'[z]) = \max_i(\deg h_i'[z]) = \max_i(\deg(a_i'[z]) + e_i)$. Then $l + \deg(\vec{\mathbf{h}}(z)) = \deg(d[z]\vec{\mathbf{h}}(z)) = \deg(\vec{\mathbf{h}}'[z]) = \max_i(\deg(a_i'[z]) + e_i) = \max_i(\deg(a_i(z)) + l + e_i) = \max_i(\deg(a_i(z)) + e_i) + l$. In general, $\begin{bmatrix} D \\ N \end{bmatrix}$ minimal and $\vec{\mathbf{h}}(z) = \begin{bmatrix} D \\ N \end{bmatrix}\vec{\mathbf{a}}(z)$ implies $\deg(\vec{\mathbf{h}}(z)) = \max_i(\deg(a_i(z)) + e_i)$.

Now given

$$\vec{\mathbf{y}}[z] = \sum_{s=1}^{e} a_s \vec{\mathbf{b}}_s = \sum_{i=1}^{m}\sum_{j=1}^{e_i} \pi_+(z^{-j}\vec{\mathbf{d}}_i)k_{ij} =$$

$$\pi_+\left\{\begin{bmatrix} D \\ N \end{bmatrix}\begin{pmatrix} k_{11}z^{-1} + k_{12}z^{-2} + \ldots + k_{1e_1}z^{-e_1} \\ \vdots \\ k_{m1}z^{-1} + k_{m2}z^{-2} + \ldots + k_{me_m}z^{-e_m} \end{pmatrix}\right\} = \pi_+\left\{\begin{bmatrix} D \\ N \end{bmatrix}\begin{pmatrix} a_1(z) \\ \vdots \\ a_m(z) \end{pmatrix}\right\}.$$

The previous remark and the fact $\vec{\mathbf{y}}[z] \neq \vec{\mathbf{0}}$ implies $\deg(\vec{\mathbf{y}}[z]) = \max_i(\deg a_i(z) + e_i)$. For each $i, 1 \leq i \leq m$, define l_i to be the minimum value of j for which $k_{ij} \neq 0$, $1 \leq j \leq \infty$. Then the $\deg(\vec{\mathbf{y}}[z]) = \max_i(e_i - l_i) = \max_i(\deg(\pi_+(z^{-l_i}\vec{\mathbf{d}}_i)) = \max_{a_s \neq 0}(\deg \vec{\mathbf{b}}_s)$. □

One may argue that, due to cancellations of high order coefficients, a linear combination $\sum a_s \vec{\mathbf{b}}_s$ in $\langle \mathcal{B} \rangle_k$ may result in a vector $\vec{\mathbf{y}}[z]$ of lesser degree than $\max_{a_s \neq 0}(\deg \vec{\mathbf{b}}_s)$. The above lemma of predictable degree says this never happens when $\begin{bmatrix} D \\ N \end{bmatrix}$ is minimal.

Theorem 1 $\langle \mathcal{B}_t \rangle \cong W_t$ *as vector spaces.*

Proof: Clearly $\langle \mathcal{B}_t \rangle \subseteq W_t$. Suppose $\vec{\mathbf{x}} \neq \vec{\mathbf{0}}$ is a coset representative from W_t. By definition there exists $\vec{\mathbf{c}}(z) \in \mathcal{C} = \mathrm{im}\begin{bmatrix} D \\ N \end{bmatrix}$ such that $\pi_+(\vec{\mathbf{c}}(z)) \equiv \vec{\mathbf{x}}$ and s such that

$\pi_+(\vec{c}(z)) = \vec{k}_0 + \ldots + \vec{k}_s z^s, s \le t$. We assume w.l.o.g. that s is the least integer such that the above is true. Also $\vec{c}(z) \in \mathcal{C}$ implies there exist $a_1(z), \ldots, a_m(z) \in k(z)$ so that

$$\vec{c}(z) = \sum_{i=1}^{m} a_i(z)\vec{d}_i = \begin{bmatrix} D \\ N \end{bmatrix} \begin{pmatrix} a_1(z) \\ \vdots \\ a_m(z) \end{pmatrix}.$$

Then

$$\vec{k}_0 + \ldots + \vec{k}_s z^s = \pi_+(\vec{c}(z)) = \pi_+(\sum_{i=1}^{m} a_i(z)\vec{d}_i) =$$

$$\pi_+\left\{\begin{bmatrix} D \\ N \end{bmatrix}(\pi_+(\vec{a}) + \pi_-(\vec{a}))\right\} = \begin{bmatrix} D \\ N \end{bmatrix}(\pi_+(\vec{a})) + \pi_+\left\{\begin{bmatrix} D \\ N \end{bmatrix}(\pi_-(\vec{a}))\right\}.$$

Consider the term $\pi_+\left\{\begin{bmatrix} D \\ N \end{bmatrix}(\pi_-(\vec{a}))\right\}$. Write

$$\pi_-(a_i(z)) = A_1^i z^{-1} + A_2^i z^{-2} + \ldots + A_{e_i}^i z^{-e_i} + \ldots.$$

Note that $\vec{d}_i A_q^i z^{-q}$, for $q > e_i$, is a strictly proper vector. So that $\pi_+(\vec{d}_i A_q^i z^{-q}) = \vec{0}$. Then

$$\pi_+\left\{\begin{bmatrix} D \\ N \end{bmatrix}(\pi_-(\vec{a}))\right\} = \pi_+(\sum_{i=1}^{m} \vec{d}_i(A_1^i z^{-1} + A_2^i z^{-2} + \ldots + A_{e_i}^i z^{-e_i} + \ldots)) =$$

$$\sum_{i=1}^{m}(A_1^i \pi_+(z^{-1}\vec{d}_i) + A_2^i \pi_+(z^{-2}\vec{d}_i) + \ldots + A_{e_i}^i \pi_+(z^{-e_i}\vec{d}_i)) \overset{def}{=} \vec{w},$$

which is just a linear combination of basis vectors from $\mathcal{B}$. Also $\begin{bmatrix} D \\ N \end{bmatrix}(\pi_+(\vec{a})) \equiv \vec{0}$ in $W(\mathcal{C})$ because it is polynomial. (Note: As a corollary we have also shown that $\mathcal{B}$ must span $W(\mathcal{C})$. This proves Lemma 1(vi) except for linear independence of the elements of $\mathcal{B}$. This can be shown by noting that in equation (15) $\dim_k(W(\mathcal{C})) = \dim_k(\mathcal{X}) = \sum e_i = |\mathcal{B}|$. So $\mathcal{B}$ is a spanning set, has the right cardinality, and so must be a basis. This said, we consider Lemma 1(vi) to be proven.) By definition $\vec{w} \equiv \vec{x}$ and so $\vec{w}$ is the unique representative from $\mathcal{B}$ for $[\vec{x}] \in W(\mathcal{C})$. It remains to show $(\deg \vec{w}) \le s \le t$, which would imply $\vec{w} \in \langle \mathcal{B}_t \rangle_k$ or that $W_t \subseteq \langle \mathcal{B}_t \rangle_k$.

Define $\alpha_i = \deg(\pi_+ a_i(z)); \beta_i = \deg(\pi_- a_i(z)); m(\alpha_i, \beta_i) = \begin{cases} \alpha_i & \text{if } \alpha_i > \beta_i \\ \beta_i & \text{if } \pi_+ a_i(z) = 0 \\ -\infty & \text{if } a_i(z) = 0. \end{cases}$

Then from the lemma of predictable degree

$$s = \deg(\pi_+(\begin{bmatrix} D \\ N \end{bmatrix}\vec{a}(z))) = \deg(\begin{bmatrix} D \\ N \end{bmatrix}\vec{a}(z)) = \max_i[m(\alpha_i, \beta_i) + e_i] =$$

$$\max_i[(\alpha_i + e_i), (\beta_i + e_i)] = \max[\max_i(\alpha_i + e_i), \max_i(\beta_i + e_i)] =$$

$$\max[\deg[\begin{bmatrix} D \\ N \end{bmatrix}\pi_+\vec{a}], \deg[\pi_+\{\begin{bmatrix} D \\ N \end{bmatrix}\pi_-\vec{a}\}] = \max[\deg[\begin{bmatrix} D \\ N \end{bmatrix}\pi_+\vec{a}], \deg[\vec{w}]].$$

This implies $\deg[\begin{bmatrix} D \\ N \end{bmatrix} \pi_+ \vec{\mathbf{a}}] \leq s$ and $\deg[\vec{\mathbf{w}}] \leq s$. Then $W_t = \langle \mathcal{B}_t \rangle_k$. $\square$

The proof is constructive and illustrates how bases for spaces in the Wedderburn filtration may be built from a minimal polynomial basis matrix $\begin{bmatrix} D \\ N \end{bmatrix}$ of the space $\mathcal{C}$. This filtration produces a partition $P = \{s_i\} = \{dim_k \, (\frac{W_i}{W_{i-1}}) | i = 0, \ldots, s\}$ whose conjugate partition P' is also the set of Wedderburn Forney indices $\{e_i\}$ above.

The *Global Controllability Filtration* (GCF) is the increasing chain $\{H_i\}_{i=1}^{s}$ defined on $\mathcal{X}(G)$ via the pullback of β in Lemma 1(v) (*i.e.*, $H_i := \beta^{-1}(W_{i-1})$). From equation (18) and the definition of β we have more concretely that $H_i =$

$$\langle \{ (\pi_+(D\vec{\mathbf{1}}_j z^{-v_j}), \pi_-(D\vec{\mathbf{1}}_j z^{-v_j})) | j = 1, \ldots, m; 1 \leq v_j = e_j, \ldots, e_j - i + 1\} \rangle_k. \quad (19)$$

where $\vec{\mathbf{1}}_j$ is the standard basis vector with 1 in the j^{th} position and zeros elsewhere.

4 Support for Definitions of GCF and WF

Case I: In the classical case G is strictly proper and $X_\infty(G) = 0$. Lemma 1(v) shows $X(G) \overset{\beta}{\cong} W(\mathrm{im}\begin{bmatrix} D \\ N \end{bmatrix})$. Then $\{H_i\}_{i=1}^{s}$ is just the *classical controllability filtration* (CCF). That is $H_i = \langle [B, AB, \ldots, A^{i-1}B] \rangle_k \overset{\beta}{\cong} W_{i-1}$, the conjugate partition of the WF gives the classical controllability indices, and the GCF is just the CCF . This can be seen, for example, by observing that the number of X's in the i^{th} row of the Kailath crate construction equals $\dim(\frac{H_i}{H_{i-1}}) = \dim(\frac{W_{i-1}}{W_{i-2}})$, see [5, 6].

Case II: When G is polynomial, $X(G) = 0$ and so $W(\mathrm{im}\begin{bmatrix} I \\ G \end{bmatrix}) \overset{\beta}{\cong} X_\infty(G) \cong [JB, \ldots, J^{s-1}B]$. In this case $H_i = (\#$ linearly independent eigenvectors of J of length $\leq i) = \ker J^i \cong W_{i-1}$. Again the spaces match and the conjugate partitions of the WF and GCF give the controllability indices, See [1].

Case III: For an arbitrary transfer function G, Malabre in [7] defined the following filtration, which fills up $\mathcal{X}'$:

$$R^0 = \mathcal{X}' \cap \ker \hat{E} \quad (20)$$

$$R^{\mu+1} = \mathcal{X}' \cap \hat{E}^{-1}(\hat{B} + \hat{A}R^\mu). \quad (21)$$

If we view $\hat{E}$ as in equation (3), it can be shown from equations 9 through 12 that

$$X'_\infty(G) \oplus X(G) \xrightarrow{\hat{E}} JX'_\infty(G) \oplus X(G) =$$

$$z^{-1}X'_\infty(G) \oplus X(G) = X_\infty(G) \oplus X(G) = \mathrm{im}\hat{E}. \quad (22)$$

Then

$$R^\mu = \hat{E}^{-1}[\hat{B} + \hat{A}R^{\mu-1}] \xrightarrow{\hat{E}} ([\hat{B} + \hat{A}R^{\mu-1}] \cap \mathrm{im}\hat{E}) \overset{def}{=} \hat{E}R^\mu \subseteq X_\infty(G) \oplus X(G). \quad (23)$$

By this map the image under $\hat{E}$ of the Malabre filtration induces an increasing filtration $\hat{E}R^\mu$ that fills up the global pole space $\mathcal{X}(G)$.

Theorem 2 $\hat{E}R^{\mu} = H_{\mu} \overset{\beta}{\cong} W_{\mu-1},\ \mu = 1, 2, \ldots, e_1.$

Sketch of Proof: (Induction) We show $\hat{E}R^{\mu} = H_{\mu}$. Note the action of $\hat{A}$ is represented by the module operator $(z, 1)$ and the action of $\hat{E}$ is represented by the module operator $(1, z^{-1})$ on the module structure of $X(G) \oplus X_{\infty}(G)$.

$$H_1 \overset{def}{=} \langle \{ (\pi_+(D\vec{\mathbf{1}}_j z^{-e_j}), \pi_-(D\vec{\mathbf{1}}_j z^{-e_j})) \} \rangle_k, j = 1, 2, \ldots, m.$$

and

$$\hat{E}R^1 = \ker \hat{E} + \hat{B}.$$

It suffices to check our result on basis vectors; w.l.o.g. let $j = 1$ and

$$\vec{\mathbf{x}} = \vec{\mathbf{d}}_1 z^{-e_1} = \vec{\mathbf{u}}_{e_1} + \vec{\mathbf{u}}_{e_1 - 1} z^{-1} + \ldots + \vec{\mathbf{u}}_0 z^{-e_1} \in H_1.$$

From equation (22), $\vec{\mathbf{d}}_1 z^{e_1} \in \mathcal{X}(G)$ implies $\vec{\mathbf{d}}_1 z^{e_1} \in \text{im}\hat{E}$. Rewrite $\vec{\mathbf{x}}$ as

$$(\vec{\mathbf{u}}, \vec{\mathbf{v}}) \oplus (\vec{\mathbf{0}}, \vec{\mathbf{v}}(z)) \overset{def}{=} (\vec{\mathbf{u}}_{e_1}, -\vec{\mathbf{u}}_{e_1}) \oplus (\vec{\mathbf{0}}, \vec{\mathbf{u}}_{e_1} + \vec{\mathbf{u}}_{e_1 - 1} z^{-1} + \ldots + \vec{\mathbf{u}}_0 z^{-e_1}).$$

The vectors $\vec{\mathbf{u}}_{e_1}$ and $-\vec{\mathbf{u}}_{e_1}$ have state space representatives given, respectively, as $B_0 \vec{\mathbf{u}}_{e_1} \in X(G)$. and $-B_{\infty}(-\vec{\mathbf{u}}_{e_1}) \in X'_{\infty}(G)$, see [1]. Then in $\mathcal{X}(G)$, $(\vec{\mathbf{u}}_{e_1}, -\vec{\mathbf{u}}_{e_1})$ has state space representative $\hat{B}\vec{\mathbf{u}}_{e_1} \in \text{im}\hat{B}$. Call this result A.
Also

$$\hat{E}(\vec{\mathbf{0}}, \vec{\mathbf{v}}(z)) = (1, z^{-1})(\vec{\mathbf{0}}, \vec{\mathbf{u}}_{e_1} + \vec{\mathbf{u}}_{e_1 - 1} z^{-1} + \ldots + \vec{\mathbf{u}}_0 z^{-e_1}) = (\vec{\mathbf{0}}, z^{-1}\vec{\mathbf{u}}_{e_1} + \ldots + \vec{\mathbf{u}}_0 z^{-(e_1+1)}) =$$

$$(\vec{\mathbf{0}}, \vec{\mathbf{d}}_1 z^{-(e_1+1)}) \xrightarrow{\beta} \pi_+(\begin{bmatrix} D \\ N \end{bmatrix} D^{-1} \vec{\mathbf{d}}_1 z^{-(e_1+1)}) = \pi_+(\begin{bmatrix} D \\ N \end{bmatrix} \vec{\mathbf{1}}_1 z^{-(e_1+1)}) \cong \vec{\mathbf{0}}$$

in $W(\text{im}\begin{bmatrix} I \\ G \end{bmatrix})$.
Because β is an isomorphism, $(0, \vec{\mathbf{v}}(z)) \in \ker \hat{E}$. Call this result B. Results A and B imply $H_1 \subseteq \hat{E}R^1$. Proof of the reverse inclusion is omitted but follows similar methods. Then $\dim \hat{E}R^1 = \dim H_1$ so $H_1 = \hat{E}R^1$.
Induction step. Suppose

$$\vec{\mathbf{d}}_1 z^{-e_1 + i - 1} \in \hat{E}R^i = (\hat{A}R^{i-1} + \hat{B}) \cap \hat{E}.$$

We wish to show

$$\vec{\mathbf{d}}_1 z^{-e_1 + i} \in \hat{E}R^{i+1} = (\hat{A}R^i + \hat{B}) \cap \hat{E}.$$

(Again we have from equation (22) that $\vec{\mathbf{d}}_1 z^{-e_1 + i} \in \text{im}\hat{E}$.) Now

$$\vec{\mathbf{d}}_1 z^{-e_1 + i} = \vec{\mathbf{u}}_{e_1} z^i + \vec{\mathbf{u}}_{e_1 - 1} z^{i-1} + \ldots + \vec{\mathbf{u}}_{e_1 - i} z^0 + \vec{\mathbf{u}}_{e_1 - i - 1} z^{-1} + \ldots + \vec{\mathbf{u}}_0 z^{-e_1 + i},$$

which we rewrite as

$$(\vec{\mathbf{u}}_{e_1 - i}, -\vec{\mathbf{u}}_{e_1 - i}) + (z, 1)[(\vec{\mathbf{u}}_{e_1} z^{i-1} + \ldots + \vec{\mathbf{u}}_{e_1 - i + 1}, \vec{\mathbf{0}}) \oplus (\vec{\mathbf{0}}, \vec{\mathbf{u}}_{e_1 - i} + \ldots + \vec{\mathbf{u}}_0 z^{-e_1 + i})].$$

This is of the form im$\hat{B} + \hat{A}(*)$. It remains to show $*$ is in R^i. But

$$(1, z^{-1})(*) = (\vec{\mathbf{u}}_{e_1} z^{i-1} + \ldots + \vec{\mathbf{u}}_{e_1-i+1}, \vec{\mathbf{0}}) \oplus (1, z^{-1})(\vec{\mathbf{0}}, \vec{\mathbf{u}}_{e_1-i} + \ldots + \vec{\mathbf{u}}_0 z^{-e_1+i}) =$$

$$(\vec{\mathbf{u}}_{e_1} z^{i-1} + \ldots + \vec{\mathbf{u}}_{e_1-i+1}, \vec{\mathbf{0}}) \oplus (\vec{\mathbf{0}}, \vec{\mathbf{u}}_{e_1-i} z^{-1} + \vec{\mathbf{u}}_{e_1-i-1} z^{-2} + \ldots + \vec{\mathbf{u}}_0 z^{-e_1+i-1}) = \vec{\mathbf{d}}_1 z^{-e_1+i-1}$$

which is in $\hat{E}R^i$ by assumption. So

$$\hat{E}(*) \in (\hat{A}R^{i-1} + \hat{B}) \cap \hat{E} \Rightarrow$$

$$(*) \in \hat{E}^{-1}([\hat{A}R^{i-1} + \hat{B}] \cap \hat{E}) \subseteq \hat{E}^{-1}([\hat{A}R^{i-1} + \hat{B}] = R^i.$$

□

The image of the Malabre Controllability filtration agrees with the GCF, which is isomorphic as filtered vector spaces to the Wedderburn filtration via β.

Case IV: If we define $\mathcal{C} = \mathrm{im}\begin{bmatrix} I \\ G \end{bmatrix}$ and $\mathcal{C}^\perp = \ker[I|G^\top]$ then from equation (15) it can be shown that $W(\mathrm{im}\ \mathcal{C}^\perp) \cong W(\mathrm{im}\ \mathcal{C})$. In [5] it was shown there exists a non-singular pairing $W(\mathrm{im}\ \mathcal{C}^\perp) \times W(\mathrm{im}\ \mathcal{C}) \to k$. The orthogonal compliments of the WF induced by $\times$, namely $\{W_i^\perp\}_{i=0}^s \subseteq W(\mathcal{C}^\perp)$, form a decreasing filtration called the *Global Observability Filtration* (GOF). This GOF is such that in the classical case when G is strictly proper $W_i^\perp \cong [C^\top | A^\top C^\top | \ldots | A^{(s-i-1)\top} C^\top]^\top$ and so agrees with the classical observability filtration. The conjugate filtration of the chain of quotients, as in the definition of P in Section 3, yields the observability indices or row degrees of $[D|N]$; where $G = D^{-1}N$ and (D, N) is a L(*Left*)GCMFD [5].

5 Current Work and Conclusions

These four cases strongly support the "correctness" of the above filtration definitions and the need for more study of the Wedderburn construction. Both Wedderburn filtrations agree with and extend the classical controllability and observability chains. Presently, we are constructing a vector space definition for the global observability filtration that uses knowledge only of $(\hat{E}, \hat{A}, \hat{B}, \hat{C})$ and is analogous to Malabre's treatments for the generalized Controllability filtration.

References

[1] G. Conte, A. Perdon, "Generalized State Space Realizations of Non-proper Transfer Functions," *Systems and Control Letters*, 1, 1982.

[2] G. Conte, A.-M. Perdon, "Infinite Zero and Infinite Pole Module," *VII Int. Conf. Analysis and Opt. of Systems, Nice, Lec. Notes Control and Inf. Science*, Springer 62, pp. 302-315, 1984.

[3] B. F. Wyman, M. Sain, G. Conte, A.-M. Perdon, "Poles and Zeros of Rational Matrix Functions," *Linear Algebra and Applications*, 157 : 113-139, 1991.

[4] B. F. Wyman, M. Sain, G. Conte, A.-M. Perdon, "On the Zeros and Poles of a Transfer Function," Special Issue on Systems and Control, *Linear Algebra and Applications*, 122/123/124 : 123-144, 1989.

[5] S. J. Giust, B. F. Wyman, "Global Observability Filtrations for Improper Transfer Functions," *Proc. SINS Int. Symp. on Implicit and Nonlinear Systems*, Fort Worth, TX, Dec. 1992.

[6] T. Kailath, *Linear Systems*, Prentice-Hall: Englewood Cliffs, 1980.

[7] M. Malabre, "Geometric Characterization of 'Complete Controllability Indices' for Singular Systems," *Systems and Control Letters*, 9, 323-327, 1987.

[8] G. D. Forney, Jr. "Minimal Bases of Rational Vector Spaces with Applications to Multivariable Linear Systems," *SIAM J. Control*, 13, 493-520, 1975.

[9] R. E. Kalman, P. Falb, and M. Arbib, *Topics in Mathematical System Theory,* McGraw-Hill, New York, 1969, Chapter X.

[10] R. E. Kalman, "Algebraic Structure of Linear Dynamic Systems. I. The Module of Sigma," *Proc. Nat. Acad. Sci. U.S.A.* 54:1503-1508 (1965).

[11] B. Wyman, G. Conte, and A. Perdon, "Local and Global Linear System Theory" in *Frequency Domain and State Space Methods for Linear Systems* (C. Byrnes and A. Lindquist, Ed.), North-Holland, Amsterdam, 1986, pp. 165-181.

[12] B. Wyman and M. K. Sain, The Zero Module and Essential Inverse Systems, *IEEE Trans. Circuits and Systems* CAS-28:112-126 (1981).

[13] S. J. Giust, M. Rakowski, B. Wyman, C. Schrader, "Controllability Indices and Wedderburn-Forney Spaces," 32nd IEEE Conference on Decision and Control, San Antonio, TX, December 1993.

Factorization of (Σ_2, Σ_1)-Lossless Functions: The Continuous Time Case

Marek Rakowski

Department of Mathematics
The Ohio State University
Columbus, OH 43210

Abstract

Let Σ_1 and Σ_2 be signature matrices. A transfer function W takes (Σ_2, Σ_1)-coisometric values on the imaginary line if $W(z)\Sigma_2 W(z)^* = \Sigma_1$ whenever $\Re e\, x = 0$. The function W is (Σ_2, Σ_1)-lossless if, in addition, $W(z)\Sigma_2 W(z)^* \leq \Sigma_1$ whenever $\Re e\, x \geq 0$ and $W(z)\Sigma_2 W(z)^* \geq \Sigma_1$ whenever $\Re e\, x \leq 0$. If (A, B, C, D) is a minimal realization of W and $\sigma(A) \cap \sigma(-A^*) = \emptyset$, we show that W takes (Σ_2, Σ_1)-coisometric values on the imaginary line if and only if there exists a hermitian matrix G which satisfies a Lyapunov equation subject to an additional constraint. This G is unique and, if $G \geq 0$, W is (Σ_2, Σ_1)-lossless. The matrix G can be used to obtain a minimal factorization $W = W_1 W_2$ where W_1 is Σ_1-lossless and W_2 is (Σ_2, Σ_1)-lossless.

1 Introduction

Let Σ be a signature matrix, that is, $\Sigma = \Sigma^* = \Sigma^{-1}$. A transfer function W of a continuous time system is Σ-lossless if

$$W(z)\Sigma W(z)^* \leq \Sigma \qquad \text{whenever} \quad \Re e\, z > 0, \tag{1.1}$$

$$W(z)\Sigma W(z)^* = \Sigma \qquad \text{whenever} \quad \Re e\, z = 0, \tag{1.2}$$

$$W(z)\Sigma W(z)^* \geq \Sigma \qquad \text{whenever} \quad \Re e\, z < 0. \tag{1.3}$$

The definition of Σ-losslessness can be extended to rectangular transfer functions as follows. Given two signature matrices Σ_1 and Σ_2 and a transfer function W, we will say that W is (Σ_2, Σ_1)-lossless if

$$W(z)\Sigma_2 W(z)^* \leq \Sigma_1 \qquad \text{whenever} \quad \Re e\, z > 0, \tag{1.4}$$

$$W(z)\Sigma_2 W(z)^* = \Sigma_1 \qquad \text{whenever} \quad \Re e\, z = 0, \tag{1.5}$$

$$W(z)\Sigma_2 W(z)^* \geq \Sigma_1 \qquad \text{whenever} \quad \Re e\, z < 0. \tag{1.6}$$

This research was partially supported by the National Science Foundation Grant DMS-9302706

If W satisfies condition (1.5), we will say that W takes (Σ_2, Σ_1)-coisometric values on the imaginary line.

Similarly as in the case $\Sigma_1 = \Sigma_2$, one can consider (Σ_2, Σ_1)-lossless transfer functions of discrete time systems. A transfer function W of a discrete time system is (Σ_2, Σ_1)-lossless if conditions (1.4)-(1.6) hold whenever $|z| > 1$, $|z| = 1$, and $|z| < 1$, respectively. Actually, the discrete time case can be reduced to the continuous time case. Indeed, if (A, B, C, D) is a realization of a transfer function R of a discrete time system, $\Theta \in [0, 2\pi]$, and $\psi(z) = e^{i\Theta}\frac{1+z}{1-z}$, then the function $W = R \circ \psi$ has a realization $(\tilde{A}, \tilde{B}, \tilde{C}, \tilde{D})$ where ([8], Appendix 1)

$$\begin{aligned} \tilde{A} &= (A - e^{i\Theta})(A + e^{i\Theta})^{-1}, & \tilde{B} &= \sqrt{2}e^{\frac{i\Theta}{2}}(A + e^{i\Theta})^{-1}B, \\ \tilde{C} &= \sqrt{2}e^{\frac{i\Theta}{2}}C(A + e^{i\Theta})^{-1}, & \tilde{D} &= D - C(A + e^{i\Theta})^{-1}B. \end{aligned} \tag{1.7}$$

Since ψ maps the left half plane onto the unit disk, W satisfies conditions (1.4)-(1.6) if and only if R satisfies corresponding conditions outside, on, and inside the unit circle, respectively. For this reason, we will consider exclusively the continuous time case.

It has been shown in [8] that a transfer function with a minimal realization (A, B, C, D) is Σ-lossless if and only if there exists a positive definite matrix P such that

$$\begin{bmatrix} AP + PA^* & PC^* \\ CP & 0 \end{bmatrix} + \begin{bmatrix} B \\ D \end{bmatrix} \Sigma \begin{bmatrix} B^* & D^* \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & \Sigma \end{bmatrix}. \tag{1.8}$$

Moreover, a Σ-lossless transfer function admits a minimal factorization into a product of Σ-lossless factors of degree 1. Below, we obtain partial extension of these results to (Σ_2, Σ_1)-lossless functions.

The paper is organized as follows. In Section 2, we characterize functions which take (Σ_2, Σ_1)-coisometric values on the imaginary line in the case where no two poles of W are symmetric with respect to the line (a more general case is addressed in [2]). This characterization provides for each minimal realization (A, B, C, D) of W a unique hermitian matrix G, called the associated hermitian matrix. In Section 3, we show that if G is positive semidefinite, then W is (Σ_2, Σ_1)-lossless. In Section 4, we use G to factor the function W into a product of a Σ_1-lossless and a (Σ_2, Σ_1)-lossless function.

2 Coisometrically valued functions

If H is a hermitian matrix, we will denote by $[\cdot, \cdot]_H$ the indefinite inner product

$$[x, y]_H = < Hx, y > = y^* Hx \tag{2.1}$$

on $\mathbf{C}^N$. If $\Sigma_1 \in \mathbf{C}^{m\times m}$ and $\Sigma_2 \in \mathbf{C}^{n\times n}$ are signature matrices, $D \in \mathbf{C}^{m\times n}$ is a (Σ_2, Σ_1)-coisometry if D^* is an isometry from $(\mathbf{C}^m, [\cdot, \cdot]_{\Sigma_1})$ to $(\mathbf{C}^n, [\cdot, \cdot]_{\Sigma_2})$. Note that D is a (Σ_2, Σ_1)-coisometry if and only if $[x, x]_{\Sigma_1} = [D^*x, D^*x]_{\Sigma_2}$ for all $x \in \mathbf{C}^m$. The last condition is equivalent to $x^*\Sigma_1 x = x^* D\Sigma_2 D^* x$ for all $x \in \mathbf{C}^m$, or $\Sigma_1 = D\Sigma_2 D^*$.

Let W be an element of $\mathcal{R}^{m\times n}$, the space of $m \times n$ rational matrix functions. The function W takes (Σ_2, Σ_1)-coisometric values on the imaginary line if and only if

$$\begin{aligned} \Sigma_1 &= W(z)\Sigma_2 W(z)^* \qquad \forall z \in i\Re \cap \operatorname{Dom} W &(2.2)\\ &= W(z)\Sigma_2 W(-\overline{z})^* \qquad \forall z \in i\Re \cap \operatorname{Dom} W. &(2.3)\end{aligned}$$

Since $W(-\overline{z})^*$ is a rational matrix function, by analytic continuation the last equality is valid for all $z \in \operatorname{Dom} W$. That is, W takes (Σ_2, Σ_1)-coisometric values on the imaginary line if and only if

$$\Sigma_1 = W(z)\Sigma_2 W(-\overline{z})^* \qquad \forall z \in \operatorname{Dom} W. \tag{2.4}$$

Example 2.1 Let $\Sigma_1 = 1$, let $\Sigma_2 = I_2$, and let $W(z) = \frac{1}{\sqrt{2}}\left[\begin{array}{cc} \frac{z-1}{z+1} & 1 \end{array}\right]$. Then

$$W(z)\Sigma_2 W(-\overline{z})^* = \frac{1}{2} + \frac{1}{2}\cdot\frac{1-z^2}{1-z^2} = \Sigma_1 \tag{2.5}$$

and W takes (Σ_2, Σ_1)-coisometric values on the imaginary line. □

If $\Sigma_1 = \Sigma_2 = I$, (Σ_2, Σ_1)-coisometries become unitary matrices. Rational matrix functions unitarily valued on the imaginary line were considered in [9] in the context of model reduction. Suppose (A, B, C) is a minimal realization. Then there exists D such that $W(z)W(-\overline{z})^* = I \; \forall z$ if and only if there exists hermitian matrices $P, Q \in \mathbf{C}^{N\times N}$ such that

$$AP + PA^* + BB^* = 0 \qquad \text{and} \qquad A^*Q + QA + C^*C = 0, \tag{2.6}$$

and $PQ = I$. Moreover, if the condition holds, then

$$D^*C + B^*Q = 0 \qquad \text{and} \qquad DB^* + CP = 0 \tag{2.7}$$

and, for any unitary D which satisfies (2.7), (A, B, C, D) is a realization of a function which takes unitary values on the imaginary line. Note that if $\sigma(A) \subset \{z : \Re e\, z < 0\}$, then the controllability and observability gramians

$$P = \int_0^\infty e^{At}BB^*e^{A^*t}dt \qquad \text{and} \qquad Q = \int_0^\infty e^{A^*t}C^*Ce^{At}dt \tag{2.8}$$

are the unique solutions of equations (2.6).

If $\Sigma_1 = \Sigma_2 = \Sigma$, (Σ_2, Σ_1)-coisometrically valued functions become Σ-unitarily valued functions. Such functions were considered in [1]. If (A, B, C, D) is a minimal realization of W, TFAE

(i) W takes Σ-unitary values on the imaginary line,

(ii) $D\Sigma D^* = \Sigma$ and there exists an invertible hermitian matrix solution H of the Lyapunov equation

$$A^*X + XA = -C^*\Sigma C \qquad \text{such that} \qquad B = -H^{-1}C^*\Sigma D, \tag{2.9}$$

(iii) $D\Sigma D^* = \Sigma$ and there exists an invertible hermitian matrix solution G of the Lyapunov equation

$$XA + XA^* = -B\Sigma B^* \qquad \text{such that} \qquad C = -D\Sigma B^* G^{-1}, \tag{2.10}$$

Moreover, if (i) holds, the matrices G and H are unique and $G = H^{-1}$.

Condition (ii) above has the following analogue for (Σ_2, Σ_1)-coisometrically valued functions (see [2] for the proof). The *defect* ([7], [10]) of a rational matrix function W is the difference between the sum of multiplicities of all the poles and the sum of multiplicities of all the zeros of W.

Proposition 2.2 *If $\Sigma_1 \in \mathbf{C}^{m\times m}$ and $\Sigma_2 \in \mathbf{C}^{n\times n}$ are signature matrices, and a function $W \in \mathcal{R}^{m\times n}$ has a minimal realization (A, B, C, D), TFAE*

(i) *$D\Sigma_2 D^* = \Sigma_1$ and there exists an invertible hermitian matrix solution H of the Lyapunov equation*

$$A^*X + XA = -C^*\Sigma_1 C \qquad \textit{such that} \qquad B = -H^{-1}C^*\Sigma_1 D, \tag{2.11}$$

(ii) *W takes (Σ_2, Σ_1)-coisometric values on the imaginary line and the defect of W equals 0,*

(iii) *$W(z) = \widehat{W}(z)U$ where $\widehat{W}$ takes Σ_1-unitary values on the imaginary line and U is a (Σ_2, Σ_1)-coisometry.*

Thus, the analogue of condition (2.9) yields a simple extension of the regular case. The analogue of (2.10) leads to a more general condition, which we prove in the case where $\sigma(A) \cap \sigma(-A^*) = \emptyset$.

Theorem 2.3 *Let Σ_1 and Σ_2 be signature matrices, let (A, B, C, D) be a minimal realization of a transfer function W, and suppose $\sigma(A) \cap \sigma(-A^*) = \emptyset$. Then W takes (Σ_2, Σ_1)-coisometric values on the imaginary line if and only if $\Sigma_1 = D\Sigma_2 D^*$ and there exists a hermitian solution G of the Lyapunov equation*

$$AX + XA^* = -B\Sigma_2 B^* \qquad \textit{such that} \qquad CG = -D\Sigma_2 B^*. \tag{2.12}$$

Moreover, this G is unique.

Proof Suppose W takes (Σ_2, Σ_1)-coisometric values on the imaginary line. Then

$$\begin{aligned} \Sigma_1 \equiv W(z)\Sigma_2 W(-\overline{z})^* \equiv D\Sigma_2 D^* + \\ + \begin{bmatrix} C & -D\Sigma_2 B^* \end{bmatrix} \left(z - \begin{bmatrix} A & -B\Sigma_2 B^* \\ 0 & -A^* \end{bmatrix} \right)^{-1} \begin{bmatrix} B\Sigma_2 D^* \\ C^* \end{bmatrix}. \end{aligned} \tag{2.13}$$

Since $D\Sigma_2 D^*$ is the value of $W(z)\Sigma_2 W(-\overline{z})^*$ at infinity, $\Sigma_1 = D\Sigma_2 D^*$ and the last term in (2.13) is identically zero. Since $\sigma(A) \cap \sigma(-A^*) = \emptyset$, equation $AX + XA^* = -B\Sigma_2 B^*$

has a hermitian solution G (see Appendix 1 in [3]). Applying to the identically zero term in (2.13) a similarity transformation, we obtain

$$\begin{bmatrix} C & -CG - D\Sigma_2 B^* \end{bmatrix} \left(z - \begin{bmatrix} A & 0 \\ 0 & -A^* \end{bmatrix} \right)^{-1} \begin{bmatrix} GC^* + B\Sigma_2 D^* \\ C^* \end{bmatrix} \equiv 0. \tag{2.14}$$

Hence

$$C(z-A)^{-1}(GC^* + B\Sigma_2 D^*) \equiv (CG + D\Sigma_2 B^*)(z + A^*)^{-1} C^*. \tag{2.15}$$

Since $\sigma(A) \cap \sigma(-A^*) = \emptyset$, (2.14) is an entire function with zero value at infinity. Thus, the function (2.14) is identically zero and, by observability of the pair (C, A), $CG + D\Sigma_2 B^* = 0$.

Conversely, suppose D is a (Σ_2, Σ_1)-coisometry and G satisfies equations (2.12). Then

$$\Sigma_1 - W(z)\Sigma_2 W(-\bar{z})^* \equiv \begin{bmatrix} -C & D\Sigma_2 B^* \end{bmatrix} \left(z - \begin{bmatrix} A & -B\Sigma_2 B^* \\ 0 & -A^* \end{bmatrix} \right)^{-1} \begin{bmatrix} B\Sigma_2 D^* \\ C^* \end{bmatrix}$$

$$\equiv \begin{bmatrix} -C & 0 \end{bmatrix} \left(z - \begin{bmatrix} A & 0 \\ 0 & -A^* \end{bmatrix} \right)^{-1} \begin{bmatrix} 0 \\ C^* \end{bmatrix} \equiv 0$$

and W takes (Σ_2, Σ_1)-coisometric values on the imaginary line. □

Note that G in Theorem 2.3 satisfies equation

$$\begin{bmatrix} AG + GA^* & GC^* \\ CG & 0 \end{bmatrix} + \begin{bmatrix} B \\ D \end{bmatrix} \Sigma_2 \begin{bmatrix} B^* & D^* \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & \Sigma_1 \end{bmatrix} \tag{2.16}$$

(cf. equation (1.8)). Also, if all eigenvalues of A are in the left half plane, a condition equivalent to that in Theorem 2.3 was given in [14], Lemma 1.

3 Lossless functions

A function which takes (Σ_2, Σ_1)-coisometric values on the imaginary line is (Σ_2, Σ_1)-lossless if it satisfies conditions (1.4) and (1.6). An instance of a (Σ_2, Σ_1)-lossless function is given in Example 2.1. Indeed, since

$$\left| \frac{z-1}{z+1} \right| \begin{cases} < 1, & \text{if} \quad \Re e\, z > 0, \\ > 1, & \text{if} \quad \Re e\, z < 0, \end{cases} \tag{3.1}$$

the function W in Example 2.1 is (Σ_2, Σ_1)-lossless. We give now two other examples of (Σ_2, Σ_1)-lossless functions.

Example 3.1 Let $\Sigma_1 = \Sigma_2 = \Sigma$, and suppose $u^* \Sigma u > 0$. Let $P = (1/u^* \Sigma u) u u^*$, and let

$$W(z) = I - P\Sigma + \frac{z - \lambda}{z + \bar{\lambda}} P\Sigma \tag{3.2}$$

for some point λ in the right half-plane. Then

$$\Sigma - W(z)\Sigma W(z)^* = \left(1 - \left|\frac{z-\lambda}{z+\bar{\lambda}}\right|^2\right) P. \tag{3.3}$$

Since $x^*Px \geq 0$ for all $x \in \mathbf{C}^n$, W is Σ-lossless. □

Note that P and Σ in Example 3.1 satisfy

$$(P\Sigma)^2 = \frac{1}{(u^*\Sigma u)^2} uu^*\Sigma uu^*\Sigma = P\Sigma. \tag{3.4}$$

Thus, $P\Sigma$ is a projection, and $I - P\Sigma$ is a complementary projection. The function (3.2) is called a Blaschke-Potapov factor. Also, note that Example 3.1 gives a Σ-lossless function whenever $(u^*\Sigma u) \cdot \Re e\, \lambda > 0$.

Example 3.2 Let W_L and W_R be two Σ_1-lossless functions (e.g. as in Example 3.1), let

$$\Sigma_2 = \begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_1 \end{bmatrix}, \tag{3.5}$$

and let $W = \frac{1}{\sqrt{2}}\begin{bmatrix} W_L & W_R \end{bmatrix}$. Then

$$W(z)\Sigma_2 W(z)^* = \frac{1}{2}(W_L(z)\Sigma_1 W_L(z)^* + W_R(z)\Sigma_1 W_R(z)^*) \tag{3.6}$$

and W is (Σ_2, Σ_1)-lossless. □

If $\Sigma_1 = \Sigma_2 = \Sigma$, it has been pointed out in [8] that condition (1.3) follows from (1.1) and (1.2). In fact, any of the conditions (1.1)-(1.3) can be removed.

Proposition 3.3 *Let Σ be a signature matrix, and let W be a transfer function. If any two of the conditions* (1.1)-(1.3) *hold, then the third one is automatically satisfied.*

Proof Since Σ is a signature matrix, there exists a unitary matrix U such that $U\Sigma U^* = \operatorname{diag}(I_p, -I_q)$. After replacing $W(z)$ by $UW(z)U^*$, we may assume $U = I$. Partition W as $[W_{ij}]_{i,j=1}^2$ with $W_{11} \in \mathcal{R}^{p\times p}$. If (1.1) or (1.2) holds, then there exists point λ in the closed right half plane such that $W(z)$ is analytic at $z = \lambda$ and

$$W_{22}(\lambda)W_{22}(\lambda)^* \geq I_q + W_{21}(\lambda)W_{21}(\lambda)^*. \tag{3.7}$$

Hence $\det W_{22} \not\equiv 0$ and we can define the function

$$R(z) = \begin{bmatrix} W_{11}(z) - W_{12}(z)W_{22}(z)^{-1}W_{21}(z) & W_{12}(z)W_{22}(z)^{-1} \\ -W_{22}(z)^{-1}W_{21}(z) & W_{22}(z)^{-1} \end{bmatrix}. \tag{3.8}$$

By Theorem 16 in [8], conditions (1.1)-(1.3) are equivalent to

$$R(z)R(z)^* \leq I \qquad \text{if} \quad \Re e\, z > 0, \tag{3.9}$$

$$R(z)R(z)^* = I \qquad \text{if} \quad \Re e\, z = 0, \tag{3.10}$$

$$R(z)R(z)^* \geq I \qquad \text{if} \quad \Re e\, z < 0, \tag{3.11}$$

respectively. Suppose (3.10) holds. Then $R(-\overline{z})^* = R(z)^{-1}$ and hence conditions (3.9) and (3.11) are equivalent. Suppose conditions (3.9) and (3.11) hold. Then, for each constant vector x, $x^*R(z)R(z)^*x \leq x^*x$ if $\Re e\, z > 0$ and $x^*R(z)R(z)^*x \geq x^*x$ if $\Re e\, z < 0$. By the continuity of the function $x^*R(z)R(z)^*x$, $x^*R(z)R(z)^*x = x^*x$ if $\Re e\, z = 0$ and (3.10) holds. □

As the next example shows, Proposition 3.3 is no longer valid when $\Sigma_1 \neq \Sigma_2$ (see Example 3.8 in [2]).

Example 3.4 Let $W(z) = [\, z^{-2} \quad 1 \quad \sqrt{2}z^{-1} \quad 1\,]$, let $\Sigma_1 = [\,1\,]$, and let

$$\Sigma_2 = \begin{bmatrix} 0 & -1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{3.12}$$

Then $\Sigma_1 - W(z)\Sigma_2 W(z)^* = (z^{-1} + \overline{z}^{-1})^2$ and so W satisfies (1.4) and (1.5), but not (1.6). □

We give now a sufficient condition for (Σ_2, Σ_1)-losslessness.

Proposition 3.5 *Suppose $W \in \mathcal{R}^{m\times n}$ takes (Σ_2, Σ_1)-coisometric values on the imaginary axis, (A, B, C, D) is a minimal realization of W, and G is the associated hermitian matrix. If G is nonnegative, then W is (Σ_2, Σ_1)-lossless.*

Proof We have

$$\begin{aligned} W(z)\Sigma_2 W(z)^* = D\Sigma_2 D^* + C(z-A)^{-1}B\Sigma_2 D^* + D\Sigma_2 B^*(\overline{z} - A^*)^{-1}C^* \\ + C(z-A)^{-1}B\Sigma_2 B^*(\overline{z} - A^*)^{-1}C^*. \end{aligned} \tag{3.16}$$

By Theorem 2.3,

$$\begin{aligned} \Sigma_1 - W(z)\Sigma_2 W(z)^* &= C(z-A)^{-1}GC^* + CG(\overline{z} - A^*)^{-1}C^* \\ &\quad + C(z-A)^{-1}(GA^* + AG)(\overline{z} - A^*)^{-1}C^* &(3.17) \\ &= (z + \overline{z})C(z-A)^{-1}G(\overline{z} - A^*)^{-1}C^*. &(3.18) \end{aligned}$$

Since $G \geq 0$, $xC(z-A)^{-1}G(\overline{z} - A^*)^{-1}C^*x^* \geq 0$ for all x and z and W satisfies conditions (1.4) and (1.6). □

4 Factorization

A factorization $W = W_1 W_2$ is minimal if the McMillan degree of W equals the sum of McMillan degrees of W_1 and W_2, and the number of columns of W_1 coincides with the normal rank of W (i.e., full-rank factorization without zero-pole cancellation). A condition for existence of a minimal factorization of a function $W \in \mathcal{R}^{n\times n}$ with nonsingular value at infinity has been given in [5] (see also [4]). Let $\Theta = (A, B, C, D)$ with D invertible be a realization of W, and let

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, \qquad \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}, \qquad [C_1 \quad C_2] \tag{4.1}$$

be the matrix representations of A, B, C with respect to a decomposition $X_1 \dot{+} X_2$ of the state space. Let $D = D_1 D_2$ (with D_1 and D_2 invertible), and let

$$\Theta_1 = (A_{11}, B_1 D_2^{-1}, C_1, D_1) \quad \text{and} \quad \Theta_2 = (A_{22}, B_2, D_1^{-1} C_2, D_2). \tag{4.2}$$

Then $\Theta = \Theta_1 \Theta_2$ if and only if

$$A(X_1) \subset X_1 \qquad \text{and} \qquad A^{\times}(X_2) \subset X_2 \tag{4.3}$$

where $A^{\times} = A - BD^{-1}C$. In particular, if the realization Θ is minimal, this gives a minimal factorization of W, and each minimal factorization of W can be obtained in this way.

Generalizations of this result to singular matrix valued functions were obtained in [6], [13], and [11] (see also [12]). In [11], factorizations of W are considered under the assumption that W has neither a pole nor zero at infinity (this is equivalent to invertibility of $W(\infty)$ if W is square and its determinant does not vanish identically). Suppose (A, B, C, D) is a realization of such a W, and let $D = D_1 D_2$ be a full-rank factorization. Pick one-sided inverses D_1^L and D_2^R of D_1 and D_2, let

$$\Theta_1 = (A_{11}, B_1 D_2^R, C_1, D_1) \qquad \text{and} \qquad \Theta_2 = (A_{22}, B_2, D_1^L C_2, D_2), \tag{4.4}$$

and let $A^{\times} = A - BD_2^R D_1^L C$. Then $\Theta = \Theta_1 \Theta_2$ if and only if (4.3) holds and

$$B_1 D_2^R D_2 = B_1 \qquad \text{and} \qquad D_1 D_1^L C_2 = C_2. \tag{4.5}$$

Moreover, the factorization is minimal if the realization Θ is minimal, and each minimal factorization of W can be obtained in this way (cf. Remarks 3.2 and 3.3 in [11]).

This result specializes to factorization of coisometrically valued functions as follows (Theorem 4.1 in [2]). Denote by $G^{\dagger}$ the Moore-Penrose inverse of a matrix G. For a hermitian matrix G, let $M^{[\perp]}$ be the orthogonal companion of $M \subset \mathbf{C}^N$ with respect to the inner product $[\cdot,\cdot]_{G^{\dagger}}$, i.e.,

$$M^{[\perp]} = \{x \in \mathbf{C}^N : \; [x, m]_{G^{\dagger}} = 0 \quad \forall m \in M\}. \tag{4.6}$$

The space $(M, [\cdot,\cdot]_{G^\dagger})$ is non-degenerate if and only if $M \cap M^{[\perp]} = (0)$. Clearly, the last condition is equivalent to $\mathbf{C}^N = M \dot{+} M^{[\perp]}$.

Theorem 4.1 *Let Σ_1 and Σ_2 be signature matrices, and let $W \in \mathcal{R}^{m\times n}$ be a function analytic at infinity with (Σ_2, Σ_1)-coisometric values on the imaginary line. Pick a minimal realization (A, B, C, D) of W, let G be the associated hermitian matrix, and let $A^\times = A - B\Sigma_2 D^*\Sigma_1 C$. Let $M \subset \operatorname{im} G$ be a subspace non-degenerate with respect to the inner product $[\cdot,\cdot]_{G^\dagger}$, let $S = [\,S_1 \quad S_2\,]$ where S_1 contains a basis for M and S_2 a basis for $M^{[\perp]}$, and suppose*

$$S^{-1}AS = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, \qquad S^{-1}B = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}, \qquad CS = [\,C_1 \quad C_2\,]. \tag{4.7}$$

If

(i) *$A(M) \subset M$ and $A^\times(M^{[\perp]}) \subset M^{[\perp]}$,*
(ii) *$B_1\Sigma_2 D^*\Sigma_1 D = B_1$,*

then W admits a minimal factorization W_1W_2 where W_1 takes Σ_1-unitary values and W_2 takes (Σ_2, Σ_1)-coisometric values on the line. Moreover, if $D = D_1D_2$ where $D_1\Sigma_1 D_1^* = \Sigma_1$ and $D_2\Sigma_2 D_2^* = \Sigma_1$, then possible factors are given by the formulas

$$W_1(z) = D_1 + C_1(z - A_{11})^{-1}B_1\Sigma_2 D_2^*\Sigma_1 \tag{4.8}$$

and

$$W_2(z) = D_2 + D_1^{-1}C_2(z - A_{22})^{-1}B_2. \tag{4.9}$$

Conversely, if W_1W_2 is a minimal factorization of W such that W_1 takes Σ_1-unitary and W_2 takes (Σ_2, Σ_1)-coisometric values on the line, then there exists a subspace $M \subset \operatorname{im} G$ non-degenerate with respect to the inner product $[\cdot,\cdot]_{G^\dagger}$ such that conditions (i) *and* (ii) *are satisfied and W_1 and W_2 have minimal realizations* (4.8) *and* (4.9).

Theorem 4.1 has the following corollary.

Corollary 4.2 *Let Σ_1 and Σ_2 be signature matrices, and let $W \in \mathcal{R}^{m\times n}$ be a function which is analytic at infinity and takes (Σ_2, Σ_1)-coisometric values on the imaginary line. Pick a minimal realization (A, B, C, D) of W, suppose the associated hermitian matrix G is nonnegative, and let $A^\times = A - B\Sigma_2 D^*\Sigma_1 C$. Let $v \in \operatorname{im} G$ be an eigenvector of A with the corresponding eigenvalue λ such that $[v, v]_{G^\dagger} \cdot \Re e\,\lambda > 0$, and suppose*

(i) *$[A^\times x, v]_{G^\dagger} = 0$ whenever $[x, v]_{G^\dagger} = 0$,*
(ii) *$B_1\Sigma_2 D^*\Sigma_1 D = B_1$ where $B_1 = v^*G^\dagger B$.*

Then W admits a minimal factorization W_1W_2 where W_2 is (Σ_2, Σ_1)-lossless and

$$W_1(z) = I - P\Sigma_1 + \frac{z + \overline{\lambda}}{z - \lambda}P\Sigma_1 \qquad \text{with} \qquad P = \frac{Cvv^*C^*}{v^*C^*\Sigma_1 Cv} \tag{4.10}$$

is Σ_1-lossless.

Proof Since v is an eigenvector of A, $M = \mathrm{sp}\{v\}$ is invariant under A. For this M, the hypotheses of Theorem 4.1 specialize to (i) and (ii). The computation which leads from (4.8) to (4.10) is given in [2]. One can verify directly that W_1 is Σ_1-lossless (see Example 3.1), and the conclusion that W_2 is (Σ_2, Σ_1)-lossless follows from $G \geq 0$. □

References

[1] D. Alpay and I. Gohberg, Unitary Rational Matrix Functions, in *Topics in Interpolation Theory of Rational Matrix Valued Functions*, ed. I. Gohberg, OT 33, Birkhäuser Verlag, Basel, 175-222, 1988.

[2] D. Alpay and M. Rakowski, Rational Matrix Functions with Coisometric Values on the Imaginary Line, preprint.

[3] J. A. Ball, I. Gohberg and L. Rodman, *Interpolation of Rational Matrix Functions*, OT 45, Birkhäuser Verlag, Basel Boston Berlin, 1990.

[4] H. Bart, I. Gohberg and M. A. Kaashoek, *Minimal Factorization of Matrix and Operator Functions*, OT 1, Birkhäuser Verlag, Basel-Boston-Stuttgart, 1979.

[5] H. Bart, I. Gohberg, M. A. Kaashoek and P. Van Dooren, Factorization of Transfer Functions, *SIAM Journal of Control and Optimization* **18** (1980), 675-696.

[6] N. Cohen, On Minimal Factorization of Rational Matrix Functions, *Integral Equations and Operator Theory* **6** (1983), 647-671.

[7] G. D. Forney, Jr., Minimal bases for rational vector spaces, with applications to multivariable linear systems, *SIAM Journal of Control* **13** (1975), 493-520.

[8] Y. Genin, P. Van Dooren, T. Kailath, J.M. Delosme, M. Morf, On Σ-Lossless Transfer Functions and Related Questions, *Linear Algebra and Its Applications* **50** (1983), 251-275.

[9] K. Glover, All optimal Hankel-norm approximations of linear multivariable systems and their L^∞-error bounds, *International Journal of Control* **39** (1984), 1115-1193.

[10] T. Kailath, *Linear Systems*, Prentice-Hall, Englewood Cliffs, 1980.

[11] M. Rakowski, Generalized Pseudoinverses of Matrix Valued Functions, *Integral Equations and Operator Theory* **14** (1991), 564-585.

[12] M. Rakowski, Minimal Factorization of Rational Matrix Functions, *IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Applications* **CAS-39** (1992), 440-445.

[13] P. Van Dooren, Factorization of a Rational Matrix: the Singular Case, *Integral Equations and Operator Theory* **7** (1984), 704-741.

[14] F. B. Yeh and L. F. Wei, Inner-outer factorizations of right-invertible real-rational matrices, *Systems and Control Letters* **14** (1990), 31-36.

INDEX OF AUTHORS

Aazhang, B. 1
Abell, J. 119
Adams III, G. B. 1176
Agrawal, R. 1119
Akkihal, S. I. 739
Al-Dhahir, N. 1166
Al-Rumaih, R. M. 149
Allebach, J. P. 717
Alsugair, A. A. 829
Ammar, G. S. 247
Anderson, J. B. 655
Antsaklis, P. J. 64, 657
Arantes, D. S. 608
Argyropoulos, Y. 878
Azzo, A. H. 1071

Balakrishnan, V. 237
Balasubramanian, R. 717
Bambos, N. 576
Barbeau, M. 928
Barry, R. A. 545
Beke, H. W. 227
Birk, Y. 588
Black, B. A. 130
Blahut, R. E. 617
Bloch, N. 588
Boley, D. 227
Borth, D. E. 325
Bouman, C. A. 717
Boyd, S. 237
Brady, D. 54
Brailean, J. C. 473
Brassil, J. 197
Buckman, M. 1007
Buescher, K. L. 74

Calvetti, D. 247
Cassandras, C. G. 667
Chan, A. H. 653
Chang, L. F. 277
Chao, H. J. 375
Chen, P.-H. 918
Cheng, R. S. 11, 829
Chiasson, G. M. 325
Chong, E. K. P. 948
Cidon, I. 177
Cioffi, J. 1166
Codrington, C. 265
Coyle, E. J. 1176
Craig, K. 109
Crow, R. W. 748
Cruz, J. R. 968, 987
Cruz, R. L. 345
Custeau, G. 928

DeCarlo, R. A. 677
Deora, S. K. 207
Dietrich, P. 167
Drakunov, S. V. 677, 809

El Ghaoui, L. 237
Ephremides, A. 21
Erevelles, W. 119

Faisal, M. 462
Feron, E. 237
Fitz, M. P. 977
Fong, K. L. 1176
Forsyth, D. A. 1009
Frazho, A. E. 495
Freeman, R. A. 791
Fu, M. C. 958

Gallager, R. 727
Garg, V. K. 687
Geering, H. P. 1058
Gelenbe, E. 1129
Gelfand, S. B. 977
Georgiadis, L. 177, 1081, 1109
Ghosh, B. K. 819
Giallorenzi, T. R. 839
Giust , S. J. 1206
Goh, K. C. 515
Goswami, A. 1019
Grigoriadis, K. M. 495, 525
Guerin, R. 177, 1081

Haas, Z. J. 556, 868
Hadj-Alouane, N. B. 898
Hegde, M. V. 335
Helstrom, C. W. 462
Hodel, A. S. 275
Holloway, L. E. 697
Hu, J.-Q. 958
Hughes, B. 24
Hui, J. Y. 140, 365
Humblet, P. A. 545

Isidori, A. 801

Jankovic, M. 819
Johnson, D. S. 868

Jones, R. D. ... 74
Jordan, H. F. ... 566
Jordan, S. ... 187, 355
Julka, V. ... 667

Karol, M. J. ... 770
Kashyap, R. L. ... 997
Katsaggelos, A. K. ... 473
Kenney, J. B. ... 1139
Khayrallah, A. S. ... 643
Kherat, S. M. ... 495
King, L. S.-B. ... 119
Klapper, A. ... 625, 635
Kohno, R. ... 297
Kokotovic, P. V. ... 791
Konstantopoulos, T. ... 413
Ku, C.-Y. ... 187
Kulkarni, S. R. ... 83
Kumar, R. ... 707
Kumar, S. P. R. ... 878

Lafortune, S. ... 898
Lanterman, A. D. ... 462
Lee, C. C. ... 737
Lee, G.-K. ... 977
Lee, T.-H. ... 365
Lei, T. ... 483
Lemmon, M. ... 64
Lin, F. ... 898
Lin, V. S. ... 34
Ling, F. ... 287
Liu, D. ... 1056
Lund, M. A. ... 737

Magliveras, S. ... 463
Mang, X. ... 1129
Markman, I. ... 655
Massey, W. A. ... 859
Mathys, P. ... 149
McCaslin, S. ... 1149
McEliece, R. J. ... 207
Mehta, M. ... 1007
Melliar-Smith, P. M. ... 157
Memon, N. D. ... 463
Messina, M. J. ... 74
Michel, A. N. ... 1056, 1061
Milstein, L. B. ... 316
Min, P. S. ... 335
Mitra, S. K. ... 129
Mohandes, M. ... 265
Morris, J. M. ... 1007
Mortazavian, H. ... 938
Moser, L. E. ... 157
Murali, R. ... 24

Nguyen, A. ... 576

O'Rourke, T. P. ... 452
Ogier, R. ... 1091
Ozcelik, T. ... 473
Özbay, H. ... 535
Özgüner, Ü. ... 677, 809

Pahlavan, K. ... 307
Papanikolopoulos, N. P. ... 1029
Parekh, A. ... 1081
Passino, K. ... 677
Patel, R. V. ... 255
Pekcan, I. H. ... 375
Peshkin, M. ... 1019
Pessao, A. C. F. ... 608
Pickholtz, R. L. ... 316
Pieris, G. R. ... 598
Poor, H. V. ... 758
Pottie, G. J. ... 34

Radović, A. ... 1
Rajan, R. ... 1119
Rakowski, M. ... 1186, 1216
Ramanan, A. V. ... 566
Ramaswami, R. ... 780
Rao, R. R. ... 167
Rasky, P. D. ... 325
Ray-Chaudhuri, S. ... 653
Reichel, L. ... 247
Rimoldi, B. ... 44
Rodemich, E. R. ... 207
Rohrs, C. E. ... 1139
Rotea, M. A. ... 505
Rothwell, C. A. ... 1009
Rutenburg, V. ... 1091

Safonov, M. G. ... 515
Saidi, H. ... 335
Sain, M. K. ... 1196
Sasaki, G. H. ... 598
Satish, A. ... 997
Sauer, J. R. ... 566
Sawan, M. E. ... 1071
Sayed, A. ... 1166
Sayood, K. ... 463
Schimmels, J. ... 1019
Schmitz, B. E. ... 485
Schoen, G. M. ... 1058
Schwartz, S. C. ... 748
Sechrist, C. F. ... 103
Sewchand, W. ... 483
Shayman, M. A. ... 707
Shi, L. ... 958
Sidhu, I. ... 355

Sigman, K. 423
Sinha, A. S. C. 1048
Sivarajan, K. N. 887
Skeleton, R. E. 525, 1077
Smith, C. E. 1029
Snyder, D. L. 462
St-Denis, R. 928
Stevenson, R. L. 452, 485
Stiver, J. A. 64, 657
Su, W.-C. 809
Subrahmanyam, M. B. 1073
Suzuki, I. 93
Szeliski, R. 1039
Szpankowski, W. 1109

Tahernezhadi, M. 968
Tarn, A. J. 801
Tassiulas, L. 1099, 1109
Tawfik, H. 878
Telang, V. P. 1139
Todd, R. M. 987
Toker, O. 535
Tsai, J. T. 345
Tse, D. 727
Tsitsiklis, J. 727
Turner, J. S. 760
Tweedie, R. L. 404

Vakili, P. 433, 908
Valerio, M. 157
van Bavel, N. 1149
Varanasi, M. K. 849
Verdú, S. 758
Verma, M. S. 217
Vojcic, B. R. 316

Wang, I.-J. 948
Wang, K. 1056, 1061
Waxman, B. M. 395
Weiss, R. 1039
Whitt, W. 859
Williamson, D. 505
Williamson, G. A. 1156
Wilson, S. G. 839
Winters, J. H. 868
Wise, G. L. 739
Wyman, B. F. 1186

Yamashita, M. 93
Yao, D. D. 423
Yaz, E. 1079
Ye, H. 1061
Yin, G. 443
Yoo, J. 1176
Young, S. D. 687
Yu, G.-X. 433, 908

Zazanis, M. A. 918
Zegura, E. W. 385
Zhang, Q. 443
Zhang, X. 54
Zhao, Z. 483
Zhu, G. 1077
Zimmermann, S. 1156

UNIVERSITY OF ILLINOIS-URBANA

3 0112 043446241